

digit Fast Track

to Open Source software

What is open source

Open source for professionals

Open source for home use

Creating your own Linux distro

Open source distributions

YOUR HANDY GUIDE TO EVERYDAY TECHNOLOGY

**Fast Track
to
Open Source
Software**

By Team Digit

Credits

The People Behind This Book

EDITORIAL

Editor-in-chief

Edward Henning

Editor

Robert Sovereign-Smith

Head-Copy Desk

Nash David

Writers

Mohamed Rameez, Srikanth R, Nishith Rastogi,
Rahil Banthia, Aditya Madanapalle, Rossi Fernandes

DESIGN AND LAYOUT

Layout Design

Vijay Padaya, MV Sajeev

Cover Design

Kabir Malkani

© 9.9 Interactive Pvt. Ltd.

Published by 9.9 Interactive

No part of this book may be reproduced, stored in a retrieval system or transmitted in any form or by any means without the prior written permission of the publisher.

August 2009

Free with Digit. Not to be sold separately. If you have paid separately for this book, please email the editor at editor@thinkdigit.com along with details of location of purchase, for appropriate action.

CONTENTS

Chapter 1	What is Open Source	8
1.1	A history of open source	9
1.2	Open source licenses	13
1.3	How do open source projects work?	14
1.4	Limitations of open source	19
Chapter 2	Open Source for Professionals	21
2.1	Open Source for the medical profession	22
2.2	Open source for media authoring	25
2.3	Open source resources for education	27
2.4	Open Source for IT professionals	30
2.5	Open source for scientists	30
2.6	Open source for engineers	31
2.7	Open Source for design professionals	33
2.8	Open source for small and medium businesses	34
Chapter 3	Open Source for Home Use	38
3.1	Introduction	38
3.2	Acquiring the source	39
3.3	Compiling from source for Linux and Windows	40
3.4	Advanced compile options	46
3.5	Reporting bugs and getting help	47

Chapter 4	Creating your Own Linux Distro	51
4.1	Remastering a live CD	52
4.2	Respinning an installed and customised distro	66
4.3	Linux from scratch (LFS)	68
Chapter 5	Linux distros	73
5.1	Ubuntu	73
5.2	Slackware	81
5.3	OpenSUSE	86
5.4	Sabayon	94
5.5	Debian	103
5.6	Mandriva	108
5.7	PC-BSD	113
5.8	OpenSolaris	118
5.9	Gentoo	120
5.10	Fedora	123
5.11	Granular Linux	124
5.12	Other important Linux distributions	126

Introduction

Open source software is a all engaging term for a wide range of approaches to designing, licencing and selling software. There is no specific definition of what open source software is or is *not*, but by and large, the Debian Free Software Guidelines apply. *Open source* in common usage is pretty recent, something that was absent in its present form before the turn of the century. Both licenced and free software were released with their source code prior to that, and there were a number of other models – such as paying extra for the source of a software after buying it, buying just the source of the software, or the source being given for free to anyone who purchased the software. All of this could technically mean open source as the source code of a particular software is available to anyone who wants it, but that is not what open source means in common usage.

Open source software, as it is commonly understood, involves the process of the development of the software as well. The development schedule of the software, and the feature list is decided by online collaboration of people working for the projects. The bugs in the software are tracked by a community of testers, and their recommendations are taken seriously. Some software and flavours of Linux are in such active developments, that the current alpha versions of these often incorporate suggestions and fixes days after they are suggested. Once a software is released, depending on the licence, open source generally allows you to go ahead and do what you want with it. What does all of this mean to the lay user?

The way open source enthusiasts look at it, it is immaterial whether everyone actually takes apart the source, but the freedom to do so, is something that is vital to how computers are used. If you're investing in all the pieces of hardware that makes up a computer, you should have the freedom to use that particular hardware in any way you feel fit, instead of the ways instead of a few corporations deciding for you. To an open source enthusiast, such a freedom is a very fundamental one, and should be enshrined in the constitution if possible. All of this is still

philosophising; and there are two sides to a coin, as ensuring a revenue stream from a software that you have paid to develop, can be equally argued as an inalienable right of the companies that make the software. Are there any benefits for everyone who uses open source software?

Yes, there are. Open source software are licenced in a way that makes them legal to use as many copies of them as you want, and wherever you want. Open source software take up much less resources than their commercial counterparts, because they don't have any code for licencing, authentication, DRM, promoting other products, attached advertisements and the like. Most FOSS is entirely free of spyware and adware. Bug fixes are very regular, and you will not be stuck with a buggy version for a long time. Open source software and the formats they use are future proof. If you make a complicated 3D model, and save it in the source format of a commercial modelling software, and then twenty years down the line, the corporation that makes the software closes down, your files will be useless. Remember, that in that much time, the hardware that you ran the original software on will probably be hideously outdated. If your software is open source, such a thing can never happen, simply because a new wave of users will continue to maintain the software.

There will always be commercial software that are much better than the open source counterparts, and an equal number of open source software that are better than the various commercial counterparts. In the end, the decision to use them is up to the individual, and this Fast Track covers the important aspects of using open source software, that is how to build from source, modify the source, the different types of software available for professionals, and which Linux distro is ideal for your needs. Read on, and you will find out that using open source software is not as complicated or terrible as it seems. Once you are done, there are plenty of source codes available in the DVD for you to play around with.

What is Open Source

Open source – the term has been the centre stage of a David vs Goliath fight in the software community. Be it Windows vs other operating systems, or Internet Explorer vs Mozilla; and now more recently, or more recently Microsoft vs Google. open source has been the rallying point of the so called ‘resistance’ to the proprietary regime. For developers within the software community, open source has transcended the role of a mere development methodology into a core life philosophy. Today, we hear of not just open source software, but also open source hardware, open source biotechnology, even open source politics!

Open source is a methodology or approach towards the design and development of software with the intention of giving the user access to the source code; i.e. if you use open source software, not only will you be able to use it, you will also be able to see how it works, debug it, modify it and redistribute it.

To elaborate and make the difference more clear, imagine the simplest entry level C++ programme, the one that says “hello world.”

If the program was proprietary (non open source), you would merely have access to the EXE file that upon execution would give a window with “hello world” written in it.

However, if the program is open source, you will also be given access to the code that the EXE file is compiled from, namely

```
#include<iostream.h>
void main()
{
cout<<"hello world";
}
```



The open source initiative

Hence, not only will you be able to use this programme for the pleasure of hearing a “hello world” from your computer, you will be able to modify this code to allow it to say more than just that.

This example of course, was overly simplistic. Even if the programme had been proprietary, anyone could recreate it making it immaterial whether it was open source or not. However, in today’s era of programmes that run up to tens of thousands of lines of code, with complex algorithms and databases forming an integral part of the system, it is impossible to recreate the code from observing how a program functions. Here, open source makes all the difference.

The Open Source Initiative, an organisation committed to promoting open source uses the open source definition in order to determine if a software is open source or not. This definition, among other things specifies requirements of the licence such as preventing discrimination against any persons or groups and preventing discrimination against fields of endeavour.

One must not forget that there is a core difference between open source and free software. While they both embody the same core idea, the former still allows the commercial utilisation of the code with the motive of profit. Many companies such as Red Hat gain significant revenues from sale, distribution, maintenance and consultancy services provided on open source software.

1.1 A history of open source

While the term open source has been coined only recently in 1998, the core concept behind it has been existent ever since the advent of software.

In the ‘50s and ‘60s, when computers and software were confined to academic and specialised industrial environments, almost all software was essentially open source. Software was not viewed as a commodity and the source code for most programs were as freely available as the programs themselves. This was due to the fact that in those days the community of developers and users of software was more or less the same. The users often modified the code to fix bugs or to improve functionality and use it themselves.

Most operating systems were freely distributed along with

their source code, and user groups such as SHARE were formed to facilitate the sharing and exchange of IBM 701 source code.

However, by the late 1960s with the evolution of high level languages and operating systems, software development costs were increasing. While most earlier software had been bundled with the hardware it was meant to be used on, there was software in the market that was exclusively for sale only.

Some programs such as Unix were given free of cost to academia and related users by AT&T, but in the absence of the permission to modify and redistribute, this was hardly open source.

Soon, by the 1970s, there were exclusive software-only companies selling software with licences that forbade modification and redistribution through patents, copyrights and trademarks. By the 1980s, nothing was left of free software as known till then. Even Unix was marketed commercially, and open source was dead.

In 1983, Richard Stallman of MIT was annoyed at Symbolics Inc. for not allowing modifications to updates on its LISP machines based on MIT's code and started the GNU project. The GNU project aimed at creating a UNIX-like open source operating system free from regulations preventing it from being modified and redistributed.

The GNU project is also aimed at creating open source development tools such as compilers and libraries. This is the beginning of modern open source as we know it, though the term 'open source' wouldn't come into existence for another 15 years. Around the same time, Stallman also founded the free software foundation that coined the free software definition – a precursor to the open source definition.



GNU

**GNU is a
recursive
acronym for
GNU's not
Unix**

The GNU operating system kernel (the central part of an operating system that performs memory and process management), called the GNU Hurd, was continuously delayed, but most of the remaining objectives of GNU had been achieved by 1991. With the release of the Linux kernel along with its source code by Linus Torvalds under the GNU General Public License (GPL) in 1992, the remaining GNU tools were modified to run on Linux and the initial objectives of GNU were considered complete. A brief naming controversy ensued that continues even today wherein the open source community is divided on whether to call it Linux or GNU/Linux.

Through all this, the software community also witnessed the development of FreeBSD and NetBSD (Berkeley Software Distributions) as open source operating systems. This culminated in OpenBSD, released in 1995. It's development continues even today and is noted for its versatility and portability.

Since 1993, numerous operating systems based on GNU, Linux and other open source software have been released. The most popular among this was most probably Debian GNU/Linux.

In the later 1990s, some of the open source software that remained popular include the Apache web server that



Richard Stallman, the founder of GNU



Linus Torvalds, the creator of the Linux Kernel. He is a legend in the open source software community



Ubuntu is a popular GNU/Linux distribution

remains the world's most popular web server software till date.

Attempts at renaming the commercially unattractive 'free software' to something that would emphasise the commercial benefits of sharing the source code with the public while not scaring profit motives away culminated in the coining of the term "open source". This caused a small ideological split in the open source community as some (such as Stallman) felt the new name undermined the philosophical and ethical aspects of free software.

Microsoft's policy of bundling Internet Explorer with Windows

is deemed to be what brought about the demise of the web browser that once dominated the market, a practice for which Microsoft was penalised in court for 'monopolistic practices'.

With increased awareness among the software industry and users about the benefits of open source, the open source movement continued to grow. Entire companies such as Red Hat Inc. developed open source software, generating income through consultancy and solutions services.

Other notable developments in the history of open source include the development of the X Window System, KDE and



Netscape was the first to open up in the browser scene



OpenOffice.org is fast emerging as an attractive open source alternative to Microsoft Office.

DID YOU KNOW?

The birth of modern open source as we know it was in 1998, when Netscape Communications Corporation decided to release the source code of its then popular Netscape Navigator programme (this code is now found in Mozilla Firefox, among others) to the public as free software. This act prompted the free software community to rethink its philosophy to try and make free software attractive to commercial software firms.

GNOME. In 1999, Sun Microsystems released StarOffice, an open source office suite that developed into the modern OpenOffice. The year 2007 saw the Java development kit (JDK) released as open source by Sun Microsystems. In July 2009, Google announced the development of Google Chrome OS, a web-based open source operating system. Many other exciting developments await the open source community.



Early this year, Google announced its intention to develop the open source Google Chrome OS

1.2 Open source licenses

Now that we are familiar with what open source is and also with its history, how do we ensure software remains open source?

The legal instrument that regulates the use and redistribution of a computer programme is known as a software licence. Proprietary software while allowing the user to use the program, maintains that the software remains the property of the vendor and limits the uses it can be put to. Microsoft Windows, for example, forbids among other things attempts to reverse engineer it, use a copy on multiple systems and publish data related to tests among other things.

While some licenses, such as the GNU GPL makes it mandatory for the user to ensure that any derivative of the software he develops remains open source, others such as the MIT license allows the user to make the code a part of a proprietary software of his own.

Open source software come with open source licences. These licences, while regulating programme usage, also permit freedom including access to the source code and the right to modify and redistribute

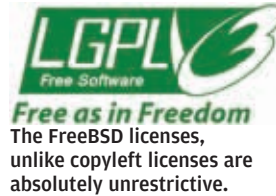
Licenses under GNU GPL, are known as 'copyleft' licenses. Other methods used by software licenses to ensure that the software remains truly open source include practices such as patent retaliation (a clause denying the user the open source freedoms if he attempts to patent any part of the software/derivative developed by him). Tivoization is aimed at

discouraging additional regulations through digital rights management. Most open source licenses restrict modified versions from claiming to be unmodified. Some also attempt to ensure that due credit is given to the copyright holders by preventing modifications of parts of the code that print warranty or license information.

The GNU General Public License and the GNU Lesser General public license. These licenses, with the famous tagline “Free as in Freedom” maintain that any derivative work/modification must be kept open source and are known as copyleft licenses.

Popular examples of open source licences include the GNU GPL, a copyleft licence that demands that any modifications/derivatives be kept copyleft. The GNU Lesser GPL, a more permissive version of the GNU GPL and the Apache licence that requires the preservation of the copyright notice and disclaimer, but is otherwise unrestricting to further uses of the source code in both proprietary and open source programmes.

The BSD license, a completely unrestricting license that allows use of the code or derivative in any way possible. Another example is the Mozilla Public License, a hybrid of the copyleft and non-restrictive licences.



1.3 How do open source projects work?

Open source projects often work in a manner completely different from traditional proprietary development. While most proprietary software is developed with fixed goals, a fixed timeline and a large amount of central control, open source development often witnesses redefinition of the goals in the middle of the development phase, forking (a new fork is when source code is taken from a project and modified independently

of the project by a separate set of developers into a different direction/into a different final product). The core concept that unifies most open source development projects is the large amount of user participation. The open source development philosophy demands that users be considered as developers, with their feedback and cooperation made paramount in debugging and further development, while they are given a chance to rectify errors and add desired features themselves.

While some open source projects start off as standalone projects for applications with a fixed goal and a minimum reliable developer network such as Mozilla Firefox, other projects such as Linux distributions (namely Debian and Ubuntu) often start off without many fixed objectives. Their objectives, components and features often emerge as development proceeds, with the final product emerging as the sum total of the additions and the views of a lot of disparate developers.

While some projects do start off with a lone developer announcing his intentions to create a software for a certain purpose in a public medium and then being joined by other interested developers, the most successful ones are the ones that start off with a developer releasing a rudimentary codebase (a very simple version of the program finally intended, often lacking in convenience features and interfaces) to the public domain under an open source license, and then other developers joining the project.

Sometimes, open source projects are also created when the source code of a considerably well developed project is released to the public for further development, as happened with Netscape Navigator.

Some organisations dedicated to a project, such as the BSD operating systems, also use a more centralised approach to development, with a central storage location for all files, daily upgraded releases and a continuous updating system.

Thus with a large development base, the core features of the open source development methodology include rapid



The success of the Mozilla project has made it an inspirational rallying point for the open source community

prototyping, incremental and evolutionary development. The large geographical distribution of the contributing development community also allows for methods such as round-the-clock development, wherein development in one way or the other proceeds at all hours of the day.

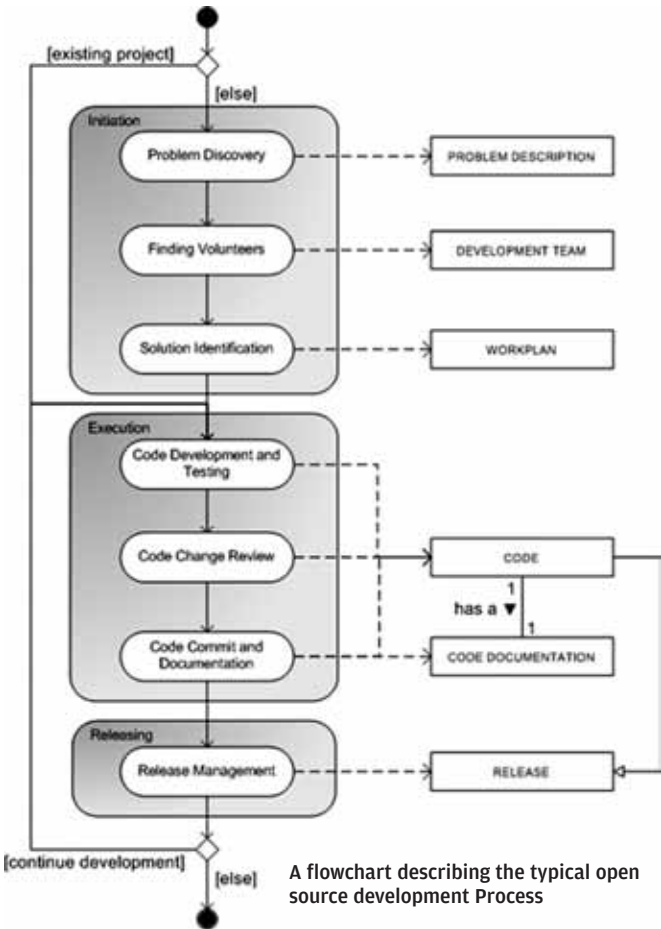
These features along with the generally larger and varied developer network often contribute towards very protracted development timelines for some popular open source projects. However, some equally important but less popular projects suffer stagnation and extremely long development times. Most open source projects continue forever, introducing continuous improvements and refinements to the code. Many projects also often have two simultaneous versions under distribution, a well tested, reliable but older version, and a new buggy and untested version with enhanced features.

The most common medium of communication that connects all the developers on a project is the internet. Web sites such as sourceforge.net strive to provide an integrated centralised communication environment for developers working on various projects. Email lists are often used to ensure that all interested parties of a project are made aware of the updates. Some projects use instant messaging through IRC or otherwise. Web forums are the most popular yet, while wikis (information database of articles maintained in the Wikipedia format with open source information sharing) provide for sharing documentation.

With a large number of developers working on a project, it often becomes difficult to identify the individual phases of development, and often version control systems are used.

The most simplistic methodology of version control, called centralised version control has all contributors to a project submitting their contributions to a centralised authority, which integrates all developments into the working whole. More recent methods such as distributed revision control rely on multiple central repositories such as mentioned in centralised version control, with various versions merged later on by merit of the quality of code, reputation of the developers, etc. The most recent tool in this field, known as the subversion revision control, is fast gaining popularity and replacing distributed revision control.

All software development projects have some sort of bug tracking system for efficient and quick debugging. While proprietary software development witnesses the developers maintaining the bug trackers among themselves during development phase, most open source projects use an internet based, accessible to all bug tracker. The most popular bug trackers among the open source community are BugZilla – a



A flowchart describing the typical open source development Process

web based sophisticated bug tracker initially used in the development of Mozilla; GNATS – a Bug Tracking system initially used by GNU; Sourceforge and its forks provide separate PHP-based bug trackers and dedicated debugging forums for all their projects.

The conventional testing process used in proprietary software development falls short of the requirements of open source development, where integration is almost continuous and changes incremental and evolutionary.

Tinderbox allows any error in the newly written code to be detected live at the time of integration, through a continuous build process. It also detects the user responsible for the erroneous update and makes it his responsibility to rectify it, overall contributing to a highly reliable debugging process. Some other features desired in debuggers for open source development include remote debugging. This feature, found in the GNU Debugger used by the GNU project allows a developer to run the debugger over the internet.

Some processes that are common in the open source development community include partial rewrites and complete rewrites. Examples include the Apache web server, which was completely rewritten between version 1.3 and 2.0.

Testing of an open source project is intrinsically easier than that of proprietary software. With the users involved from a very early stage of development, bugs are detected and corrected early during development phase, often leading to extremely stable and bug free code. In addition, automated testing routines used in proprietary programs are also used in open source projects. The fact that most open source projects through a majority of their development phase use command line interfaces or application programming interfaces also makes automated testing easier.

While the various advantages of open source development are as mentioned above, success of a project largely depends



The GNU Debugger is a popular open source debugging tool

The challengers have given rise to new debugging tools such as Tinderbox

on its popularity and the amount of volunteer developer talent it attracts. As such, publicity plays an important role in open source development, with project supporters often using directories such as freshmeat and fsf.org to promote their projects. Projects also gain publicity through articles published in open source developers community journals such as Linux Weekly News and IBM Developerworks.

For more first hand experience with open source projects, you can visit among others, www.sourceforge.net – an online source code repository which has about 1,80,000 projects and over 1.9 million registered developers.

1.4 Limitations of open source.

As with all things in life, open source development comes at a price. While open source development methodologies offer excellent solutions for developing quick, powerful and popularly needed software solutions, it is often insufficient to provide customer specific software development, related consultancy and services. The reliability of the open source development process is also under question as development timelines are notoriously prone to trends in fashion within the open source community, though the final product when produced is found to be often very reliable.

Large amounts of manpower and developer talent are wasted on some projects started new when it would have been more efficient to contribute to a pre-existing project. This effect is known as the not invented here (NIH) syndrome.

In specific areas of software development, pro-proprietary software campaigners argue that the free availability of the source code of open source software makes it easier for users with malicious objectives to find vulnerabilities in the software and exploit it.

Open source software and related businesses have been shown to be relying on a poorly constructed business model. While projects rely largely on volunteer developers and are unpredictable in terms of timelines, the companies are still answerable to their shareholders. The companies that rely on paid developers for their open source development find it difficult to reclaim their development costs through providing support and consultancy alone. Proprietary software based

business models come without these intrinsic contradictions.

The continuously incremental and evolutionary nature of open source software also ensures that most open source projects produce derivative rather than innovative products. Most of open source software in the desktop arena is also intentionally intended to harm equivalent commercial players through alternate free and open source solutions. A prime example is OpenOffice.org, intended to be an open source replacement for Microsoft Office. Another example is GIMP, essentially inspired by Adobe Photoshop.

Other arguments against open source software include the supposedly seamless way in which proprietary software such as Microsoft Office, built as a monolithic whole by a single company with its different components designed to interoperate flawlessly run. In contrast, open source software, the various components of which are developed by various programmers attempts to ensure flawless integration through the implementation of interoperability standards, many of which are often not adhered to.

Open Source for Professionals

There are many open source options available for professionals to help them in their complex tasks. The tools mentioned here come mainly in two forms – as an offline installation, or as a web service and accessible over a browser. To strike a balance, we have covered just a few in brief and offered links to a whole lot more for the inquisitive professional to look up.

While there surely is a bias towards Linux when it comes to open source, there are a few Windows-only applications. The selection of an application to review has also been influenced by the relative lack of awareness about Linux installation procedures among the general population. The Windows user can take comfort in the fact that since these Linux only applications are open source, the source code will be available and can be compiled for Windows (compiling instructions are not provided here). It needs to be noted that in the case of the very few Windows-only applications, sometimes their dependence on proprietary libraries such as the .NET framework make it impossible to run them in Linux. Emulators do exist that offer middle ground – Cygwin (www.cygwin.com) for running Linux applications in Windows and Wine (www.winehq.org) to run Windows applications in Linux.

A more general case is that of Debian, a venerable Linux distro with probably the largest package database. Debian offers different *blends* of their popular distro targeted at different pros. For example, Debian Med is a *blend* containing all packages related to the medical field. *Blending* merely involves getting the packages related to the particular field, so to create a Debian Med

A word on distros

Thanks to the freedom inherent in the open source movement, it is common to find custom live Linux distros containing several packages related to a particular profession offering a complete working environment that runs off the CD/DVD. This is significantly more convenient than finding and downloading individual applications. For example, VLinux is a complete Linux distro based on Knoppix that is available as a live CD and is targeted at bio-informatics professionals. This distro is available for download at www.bioinformatics.org/vLinux/.

blend the packages that related to the medical industry can be downloaded from their repositories (these packages are listed at <http://debian-med.alieth.debian.org/tasks/>). A *blend* for scientific pursuits (in different fields) exists and can be researched at <http://wiki.debian.org/DebianScience>. Similar distros exist for almost every field discussed in this article, and needless to say, we shall limit ourselves to only mentioning the names of such distro along with relevant links.

2.1 Open Source for the medical profession

There are a host of applications for doctors, hospital employees and all those involved in the medical field including pharmacists and medical transcriptionists as well as other medical professionals.

2.1.1 Electronic medical/health recording software

Managing patient records is a major task for any medical practitioner / hospital, not just for financial reasons, but also to readily access the medical history of patients, or to identify any trends in diseases among the population, etc.

Most of the applications in this category are web services, which means that the code needs to be installed on some webserver before it can be used. The advantage is that, if installed on a online webhost, this allows the data to be accessed with a browser from any net connected device.

A category of software catering to electronic medical records, electronic health records and practice management software has evolved

OpenEMR (www.oemr.org)

OpenEMR is an open source electronic medical record (EMR) web service. This is written in PHP and requires a web server such as Apache running MySQL in the back end. The installation is slightly complex and requires some prior exposure to installing similar applications on a web server. However, once done, the application shows potential. Every aspect of a patient's health status, and details of the interaction with the medical service provider can be recorded. Accessing the

details of previous patient visits is easily accomplished. Besides the typical medical aspects, OpenEMR also tracks the billing/payment history of patients. OpenEMR also has modules to keep track of medicines issued and the inventory status in real time. All the data can also be displayed in different reporting formats for a bird's eye view of the practice/business.

To experience this application first hand, go to www.oemr.org/modules/wiwimod/#Demos for a demo.

FreeMed (<http://www.freemed.info/freemed/index.php>)

A service similar to OpenEMR in most aspects, FreeMed is a bit simpler and probably more suited to smaller clinics.

2.1.2 Medical imaging tools

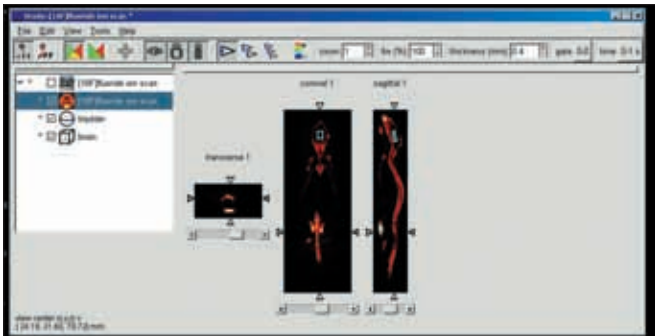
AMIDE: Amide's a Medical Imaging Data Examiner (<http://amide.sourceforge.net/index.html>)

As the name makes it clear, AMIDE is a medical imaging tool. It can handle many medical image file formats such as DICOM, which is the standard format for radiology images created from X-ray, MRI and CAT scans. AMIDE is an open source application available for Windows as a binary file and for Linux as source which need to be compiled. The feature list is long, though incomprehensible to novices. The site offers a sample set of images to try out the features of the application.

Pros looking for more options will find them in the massive

Additional resources

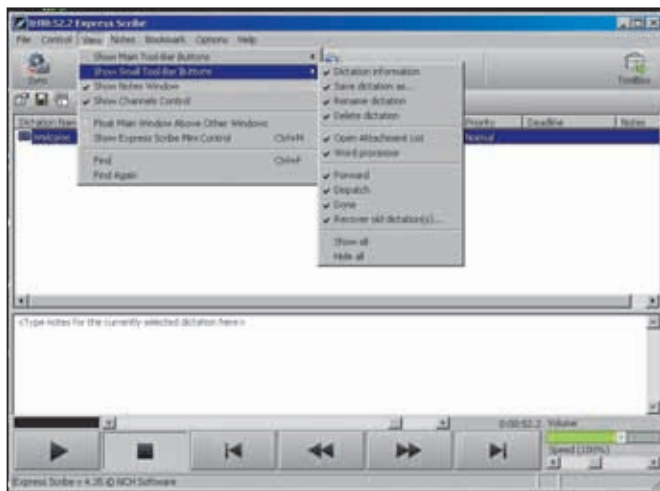
More open source EMR options are listed at <http://tkfp.sourceforge.net/links.html>



Views of the scan results of a rat (sample image) in AMIDE

Express Scribe (<http://www.nch.com.au/scribe/index.html>)

Medical transcriptionists seeking open source tools would find Express Scribe a fitting replacement for commercial applications. It's definitely worth a look, especially since it comes in a



The simple interface of Express Scribe

download package – all for 635 KB! As with other transcription software, you can also use footpedals. It incorporates a text-area where the transcribed text can be typed, though it can also be integrated into MS Word. The application is available for most operating systems. Even non-transcriptionists will find the audio playback capabilities useful, especially when listening to audio books.

Transcriber (<http://sourceforge.net/projects/trans/>)

Is another open source transcription tool that seems to have lesser features as compared to Express Scribe. However, it does its job just as well if foot pedal support is not considered.

2.2 Open source for media authoring

2.2.1 Audio editing

Traverso (<http://traverso-daw.org/>)

This is a very capable open source multi track audio editor and recording application, available for Windows, Linux and Mac users. The interface is a slightly complicated when compared to Audacity. However, on giving a closer look at it, you would realise its power. Overall, Traverso is regarded by many as an alternative to more popular commercial applications such as Sony Acid Pro and Adobe Audition.

We used Traverso on Linux. The UI is quite easy to master, with shortcut keys for all functions related to editing and mastering, and a comprehensive online help file. Traverso has also included an CD writing module, so there is no need for an external burning application to create audio CDs of the edited tracks.

Audacity (<http://audacity.sourceforge.net/>) is another open source audio editor that has an impressive gamut of functionalities as Traverso, but since it is popular and has been reviewed many times before, just a cursory mention should suffice.

Additional resources

Rosegarden (www.rosegardenmusic.com) and Ardour (www.ardour.org) are two open source audio editing tools that are not available under Windows. A long list of more audio applications is available at www.opensound.com/ossapps.html

2.2.2 Video editing

Avidemux (<http://avidemux.sourceforge.net/>)

If you've used VirtualDub (a popular open source video editing application especially for AVI files), then you'll feel right at home with Avidemux. Providing most of the functionalities of the former, Avidemux ups the ante by also allowing editing and encoding several more file types, including FLV (flash video popularly used in YouTube and other online video platforms) format.

Again, we tested Avidemux under Linux. Its capabilities range from simple video editing – cutting scenes is as simple marking them and using [Ctrl] + [X] (when we tried

pasting, we didn't get the desired result, probably due to a bug), to appending videos, to converting across video formats. Avidemux offers shortcuts to automate setting various codec options when creating content suitable for viewing on different video devices such as DVD players or PlayStation Portable; thus freeing the user from these technical details.

VirtualDub (www.virtualdub.org) is a bit more polished than Avidemux, but cannot be run natively in Linux.

2.2.3 DVD authoring

DVD styler (<http://dvdstyler.de/>)

This is an open source video DVD authoring tool that is available for both – Linux and Windows platforms. While not a professional level application, it is feature laden to help easily author video DVDs. The interface is quite user-friendly, and most tasks can be accomplished with clicks and by drag and drop.

With an integrated file browser, media files can be dragged and dropped into the time line. Clips can be shuffled and their properties modified. DVDStyler also comes preloaded with a few background images and buttons styles that can be used to create a menu, which can be further customised by the user. DVDStyler can convert videos into the default DVD standard (MPEG2) while mastering the DVD. So you don't need any prior conversion – AVI files can be directly dragged on to the time line, and are converted to MPEG2 on the fly. The final output can also be written to a DVD with the integrated burner. The application comes with a help file that is too brief, so a user is better off downloading the user guide available online to learn how to use this application at its fullest potential.



Open source aficionados will do well to take a look at other acclaimed video editing suites such as Jahshaka (<http://jahshaka.org/>), Kino (www.kinodv.org/) and Cinerella (<http://cvs.cinelerra.org/>) – the latter two are only available under Linux.

InfraRecorder (<http://infrarecorder.org/>)

While Linux users have had feature rich media burners bundled with the operating system, the one bundled with Windows is minimal. InfraRecorder, a Windows only open source

application is a good choice for people looking for open source alternatives to commercial proprietary burners. The interface and list of features offered by InfraRecorder is at par with that offered by its commercial counterparts.

2.3 Open source resources for education

2.3.1 Educational institution administration

Open admin for schools (<http://richtech.ca/openadmin/index.html>)

This is a web service that needs to be installed on a web server. You can test how this works with its demo page available on the home page. The application offers extensive data collection capabilities to monitor every aspect of a student's performance. The interface is designed to address the concerns of different types of users – school manager/principal, teachers and parents. The School manager's page allows creating and modifying different aspects of running a school such as setting fees, creating a schedule and managing enrolments.

DID YOU KNOW?

64 Studio is a Linux distro targeted at multimedia professionals. You can download this distro at www.64studio.com

Similarly, the Teacher's view allows finer control of a student's activities – recording attendance and marks. The Parents view allows viewing the progress card of the ward online. (takes the fun out of "report card day")

From a non-pro's view, the options on the pages are esoteric, but considering that the application has been accepted in many schools all over the world – on the website, among the list of schools using this application, close to 600 Indian schools are listed – a education pro may find all bases covered.

2.3.2 Course management systems

Moodle (<http://www.moodle.org>)

A courseware management tool such as Moodle is primarily used to manage an educational organisation's "product". Moodle is a web service and requires a web server to run. It's a relatively complex application probably because of the many features it sports. Some training is required for new users. Pros who have cut their teeth on content management systems, would feel right at home in Moodle.

You can access a demo of Moodle on its home page. Moodle offers distinct controls for each of its user classes – Manager, Teacher and Pupils. The Admin view, besides allowing changes to the site structure, also allows editing courseware parameters such as courses and their durations, enrolment schedules and grading. The Teacher view allows authorised teachers to create course content (or lessons), interact with users in real time through chat, create tests and quizzes and grade performance. The Student view allows users to access lessons and participate in tests created by teachers. The Chat module also allows real time interaction with the teacher.



For more open source course management systems go to www.edutools.info/feature_list.jsp?pj=4&f=823

Since the student using Moodle need not be on the campus to use the resources, Moodle is apt for distance education as well. With distance education gaining in popularity, applications such as Moodle are sure to become indispensable in the education sector.

Moodle offers a bewildering array of features and can be converted into a powerful tool to manage courseware. But, as previously stated, to achieve this users need some form of training.

2.3.3 Library management systems

Though libraries are part of an educational institution, we can include public libraries in this discussion. In fact, any lending library following the same procedure (be it books or video DVDs) can be included.

The beauty of a library management system will be appreciated when one trudges all the way to the library only to return empty handed because the last copy of a book/DVD had been lent out. If an application would allow the availability of a book/DVD to be searched from the comfort of your home, it would be good. In fact, even better would be a single portal to search for a book/DVD across many libraries! An LMS is the answer to such needs.

Evergreen (<http://www.open-ils.org>)

To appreciate the power of Evergreen, one has to only visit

the user page on the demo site. It offers a user a history of the lending habit – books lent and returned, fines paid, etc. – and a search function that can track a book across different libraries – and, even better, place a hold request in case the book is tracked down.

Evergreen is comprised of three parts. The first part is a server, where all the data is stored. This being a web service, the application has to be installed on a web server before it can be used. Unfortunately, there are no demo sites to test the server.

The second part is a “Staff Client”. This is an application used by library staff to record details of lendable items, and track patron activities. The client needs to be connected to the server for which the URL of the client and a username and password are required. The client is available in binary (or EXE) form for Windows, but has to be compiled from source for Linux. The client is minimalist, but offers the basic functions of cataloguing new books, verifying the status of existing books and managing patron relationships. Pros would be better able to appreciate the finer aspects of the options under the different menus in the Staff client. The staff client also has an offline mode that allows recording transactions without being connected to the server, which can be synchronised later after the connection is established.

The third part is the public interface. The demo site, mentioned previously, is in fact the public interface. This page can be accessed with any browser.

NewGenLib (www.verussolutions.biz/web/) is another open source application in the same category. It has been created by an Indian organisation, and is in use in major libraries in India, as culled from the web site.

Additional resources

Rosegarden (<http://www.rosegardenmusic.com/>) and Ardour (<http://www.ardour.org>) are two open source audio editing tools that are not available under Windows. A long list of more audio applications is available here <http://www.opensound.com/ossapps.html>

Additional resources

For school managements, many open source tools to organise different aspects of the working of an educational institution are listed at www.edutools.info/feature_list.jsp?pj=4&f=823.

2.4 Open Source for IT professionals

2.4.1 Integrated development environments

Eclipse (<http://www.eclipse.org/>)

Eclipse is an open source IDE that can be used for creating applications in languages such as Java, C/C++ and PHP.

However, the emphasis is on Java. Since Eclipse itself is written in Java, it will run on any system containing the Java Runtime Environment (JRE). Another popular open source Java IDE is Netbeans (www.netbeans.org).

Comparatively, there are fewer open source IDEs for non open source languages. For .NET languages, Sharpdevelop (www.icsharpcode.net/OpenSource/SD) is an open source application that can work with VB.NET and C#.

There are many open source IDEs for web based languages such as HTML, PHP and Python. Komodo Edit (www.activestate.com/komodo_edit) is one that can handle all those mentioned, and also Perl, Ruby and Tcl.

Similar is the case with tools to test web applications and web sites. A long list of such applications are listed at www.opensourcetesting.org. Universal modelling language (UML) is an part of software development. ArgoUML (<http://argouml.tigris.org>) is an open source UML modelling tool that runs in any Java environment.

2.5 Open source for scientists

2.5.1 Statistical data analysis

No research is complete without statistically testing the findings for significance. The statistical tools landscape is populated with proprietary statistical packages such as SPSS, though open source applications have started making their presence felt.

R (www.r-project.org) is a multiplatform tool for statistical analysis. R is inspired by the well known S language developed at Bell Labs in the 70s. S is now sold by the name S-plus by Tibco Software. PSPP (www.gnu.org/software/pspp) is another

statistical tool that tries to be an alternative to the commercial statistics package SPSS. There are a few more statistical applications at <https://www.msu.edu/~olsonluk/openSource/OSSMath.htm>.

Distro

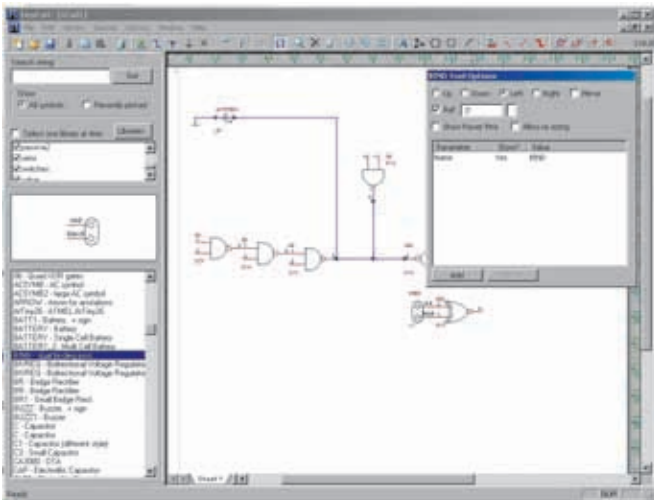
For the more scientifically and mathematically inclined, there is an interesting Linux distro available at Scientific Linux (<https://www.scientificLinux.org/>).

2.6 Open source for engineers

2.6.1 Computer Aided Design Software

TinyCAD (<http://sourceforge.net/projects/tinycad/>)

This is a niche open source CAD application that caters to printed circuit board (PCB) design. This is useful to electrical and electronics engineers. In fact, even engineering students working on their projects would find this useful. TinyCAD is available in binary form for Windows and as source for Linux. Creating circuits with TinyCAD was a breeze, with even a novice like the reviewer being able to start off with no orientation (though the circuit was not electronically logical, as it did not



Creating circuits is child's play with TinyCad

stand up to the inbuilt circuit design rules test tool). TinyCAD seems to be a competent program for the job.

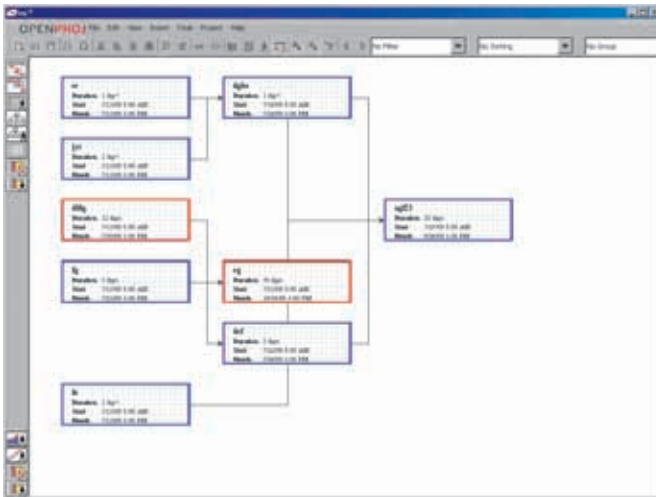
Electric VLSI design system (www.staticfreesoft.com) is another open source tool for the same purpose that is written in Java and will run on any system having the JRE installed. Electrical and electronics engineers will find many useful electronic design automation tools at <http://opencollector.org/collector.php>.

2.6.2 Project management

OpenProj (<http://openproj.org/home>)

This is an open source project management tool for all major operating systems. It purports to be a replacement for Microsoft Project. OpenProj has a very simple interface. Most professionals will appreciate the ease with which project details such as duration, priority, person in charge and resources can be recorded. It allows activity linkages to be specified with simple click and drag action. For ease of comprehension project details can be displayed in different formats such as Gantt chart, network, histogram, etc.

dotProject (www.dotproject.net) is an open source web based PM tool using PHP and MySQL, that requires the application to



Plotting events and linkages with openproj

be loaded on a web server, and can then be accessed with a web browser from any PC.

2.7 Open Source for design professionals

2.7.1 Modelling / Drawing Tools

freeCAD (<http://www.ar-cad.com/>)

This open source application is available for Linux, Windows and Mac. freeCAD is a “3d CAD with Motion Simulation” tool. Under Linux, installing freeCAD was easily accomplished.

Being a novice with design applications, the reviewer did find the menu options esoteric. But, to get things started, freeCAD has an extensive help system – every menu has an “Explain” option which describes the use of the menu.

Open source enthusiasts working in CAD and CAE environments should try out CAELinux at <http://www.caeLinux.com/CMS/>

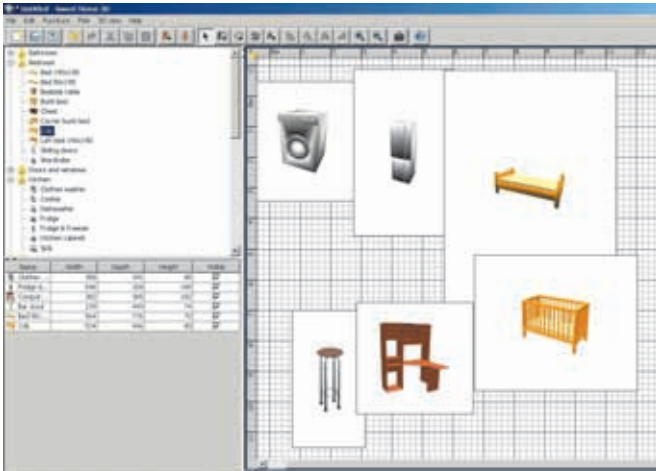
Additionally a few sample drawings are included to set the ball rolling. The interface is minimalist, to the point of being drab. But, it has the controls and features that would be appreciated by design pros. freeCAD is adware, and the ad messages are continually refreshed which can be distracting at times.

Interesting options

Blender (www.blender.org) is a popular 3D design tool, available for Linux, Windows. And, if their screenshot galleries are taken as a yardstick, opencascade (www.opencascade.org) and BRL-CAD (brlcad.org) have impressive capabilities

Sweet Home 3D (<http://www.sweethome3d.eu>)

This is probably the only open source application of its kind aimed at interior designing. The application is available for Linux and Windows, and requires JRE 5 (JRE 1.5) to run. The reviewer learnt the hard way that the application has a preference for some graphics cards. On the reviewer’s system using ProSavage onboard graphics and DirectX 9c the app would exit with an error. From the screenshots available on the site, the application seems capable of rendering home interiors quite admirably. It has built in models for furniture, building structures such as staircase and pillars. The application can also



Some ready to use 3D models in Sweet Home 3D

import 3D models of objects created in other applications, and some models are made available on the site for download. A detail how-to on the web site makes it easy for even novices to start designing a home.

2.8 Open source for small and medium businesses

The most popular open source office applications are openoffice (office suite) available at openoffice.org and Thunderbird (email client) available at www.mozilla.com/en-US/thunderbird. For a personal information manager, download Chandler at chandlerproject.org. Clamwin is an effective antivirus. You can download this at www.clamwin.com.

Those in the publishing industry can try out Scribus at www.scribus.net. Gimp is a well known image editing software pitted against the likes of Adobe Photoshop. Try it out at www.gimp.org. Inkscape is a vector drawing application available as a free download at www.inkscape.org. You can reduce your phone bills by implementing pidgin for IM. You can try it at <http://pidgin.im>.

2.8.1 Enterprise resource planning

OpenERP (<http://www.openerp.com/>)

This is an open source ERP application available for Linux

and Windows. Being free, the community edition of OpenERP offers an excellent opportunity for SMBs to avail the benefits of such a package without incurring the massive investment of their commercial counterparts. OpenERP offers different modules such as logistics, human resources management and finance to customise the installation to suit various enterprises. Demos of installations of OpenERP modified to different enterprises such as bookstores, CRM (call centres), auction house and manufacturing plants are available on the web site. A sampling of the services industry demo showed the depth of the application. With extensive recording and reporting capabilities, the application makes it easy to track every aspect of a business, and offers a bird's eye view of the organisation. Needless to say, using an ERP requires some training and the interface with its many menus and submenus can be bewildering for the novice.

The OpenERP application is available in binary form for Windows and as source code for Linux. OpenERP offers different editions of the application, with the only difference between editions being the level of support provided.

2.8.2 Accounting

Gnucash (www.gnucash.org) is high on public awareness, so we shall not wax eloquent about it. Rather we shall look at the category of web based open source accounting applications. Being web-based they need to be installed on a web server, but the benefit of being able to access one's accounts from any pc (if the webserver is online) is the upside.

Phreebooks (<http://www.phreebooks.com>)

The reason we choose this application is because it is the only one with decent features that offered a demo precluding the need to go through the web installation process. Phreebooks is written in PHP and Javascript and uses MySQL in the backend. The demo succeeds in presenting the full depth of the application. All aspects of biz accounting namely inventory, general ledger, banking and customer are covered in the interface, though it would take a pro to pass the final judgement about its applicability in a production environment.

Another online accounting solution that has an extensive feature list (but not a demo) is LedgerSMB (ledgersmb.org).



View of menus and detail listing in Phreebooks

Those SMBs who'd rather prefer offline account keeping can also check osFinancials (www.osfinancials.org) and Turbo Cash (www.turbocash.net).

2.8.3 Customer relationship management

It has become imperative for every organisation to get closer to the customer in order to survive in the market. A CRM application is just the tool to accomplish this.

vtiger (<http://www.vtiger.com>)

This open source application has Indian roots. It is a web service requiring the popular PHP, MySQL, Apache combination to install and run. vtiger is a comprehensive CRM package that



Menus for every biz activity in vtiger

allows its user to manage all aspects of a biz, namely Sales force management, Customer relations, Marketing, Inventory control etc. An online demo is available on the home page. The interface is loaded with menus and submenus which allow recording and tracking progress on different areas of the biz. But, to make any sensible use of these features some form of orientation would be required. While there is a user manual on the site, getting to it requires some dedicated clicking. vtiger also offers plugins to integrate with popular applications such as Thunderbird (email client) and MS office so that documents from these applications can be integrated with the CRM. vtiger offers an excellent mix of features and the price, as well, make it an essential application for every SMB.

Sugar CRM (www.sugarcrm.com) is another open source CRM application that is highly rated among users. The community edition of the application is free.

2.8.4 Additional Resources

For a list of open source solutions for SMBs, head to www.Linuxplanet.com/Linuxplanet/reviews/6796/1/

Conclusion

The open source movement caters to every need of a professional. The beauty of the open source scheme lies in the ability of any user to modify an existing application to better suit his/her needs, thereby creating a new application that will find takers. This and the price of the application are strong reasons to give open source applications a try.

Open Source for Home Use

3.1 Introduction

You have got the latest gear and your CPU is a water-cooled over-clocked monster satisfying your urge to live on the edge, but your software is holding you back. Either the latest and greatest lines of code are available as just “lines of code” or the available binary packages are designed for a generic environment and are not fully using the bleeding edge features of your machine. What to do? This chapter aims to provide the answer by teaching you how to compile software’s source code and install it on your machine.

As briefly mentioned in the box there are various distinct advantages of compiling the source code on your own, instead of using pre-built binary packages (exe/msi etc for Windows and deb/rpm for GNU/Linux) which are as follows:

Availability: It massively increases the range of software available for you. A lot of new and interesting software are available only as source archives, especially the ones in beta stages.

Optimisation: When you compile from source code on your own machine, the software is optimised for your specific hardware and will perform better and faster as compared to the generic built.

Bleeding edge: The daily builds of even mature software or the

Jargon buster

Before we move further it will be useful to go over a few definitions:

Source code: It is the actual human-readable code written in a language such as C, C++ or Java which is then compiled and linked by a software called “Compiler” and “Linker” (often rolled into one) to form an executable (your typical “filename.exe” file in Windows). This code can be modified to introduce, remove or modify desired features.

Binary package: This is your “setup.exe” or “filename.deb” or “filename.rpm” and similar software installer. This is machine-readable instructions and do not make sense to human eyes. Binary packages cannot be modified.

Open source software: People who write these software willingly share the source code of the same with everyone. Their prime motive to share their intellectual property with others is their belief that this open collaboration produces better results and prevents people from solving the same problem twice.

individual developer branches are generally available only as source archive and not as binary builds. Hence, if you want to test the very latest version (read unreleased) compiling from source may be your only option.

Now that we have a motive to compile our code instead of using pre-built binary packages, let us move towards the first step, which is acquiring them.

3.2 Acquiring the source

It can be a daunting task for a newcomer to merely acquire the source code and extract the actual source code tree from compressed packages. This section will cover that aspect.

A point of note over here is that in over 99 per cent cases the file you will download will be in a compressed archive but not in the ZIP format popular with Windows user, but as a tarball (.GZ or .TGZ) extension. Your Linux distribution would have come by default with a tool to handle such compressed packages.

The other method is to use a version control client.

It is often said that the art of writing good software is to keep rewriting it. That is why most software development teams use some form of version control. Cooperating developers commit their changes incrementally to a common source repository, which allows them to collaborate on code without resorting to crude file-sharing techniques (shared drives, email). Source control tools track all prior versions of all files, allowing developers to “time travel” backward and forward in their software to determine when and where bugs are introduced. These tools also identify conflicting simultaneous modifications made by two (poorly-communicating) team members, forcing them to



There are generally two ways to download the source code, one is the direct download link method and the other is using a revision control client. The first one is the more familiar method, where you go to the software's web site and spot the “download” link and chose the source archive from there.

Tip

Windows Users can use another open source tool called “7-Zip” to handle all your uncompressing needs. This is available as a free download in both binary and source format at www.7-zip.org/download.html.

work out the correct solution (rather than blindly overwriting one or the other original submission).

The two most commonly used ones are CVS and SVN.

CVS is an acronym for Concurrent Version System and SVN is for Subversion. On the download page of the software's web site you will often see CVS/SVN link, which can be accessed via CVS or SVN clients. If you are a Windows user you can use clients from Tortoise available at www.tortoisecvs.org and www.tortoissvn.net.

They will directly integrate with your "Windows Explorer" and let you download and explore the source-code tree.

If you are a Linux user you can use your favourite package manager (synaptics/yum/emerge) to download the "cvs" and the "subversion" package. These will also be installed for you in the cygwin shell. And then just use the `cvs` or `svn` command on the command line.

Example

To download the source code of TuxPaint (an MS Paint alternative) you can run the following commands:

For read-only anonymous access, type:

```
cvs -d:pserver:anonymous@tuxpaint.cvs.  
sourceforge.net:/cvsroot/tuxpaint login  
and then
```

```
cvs -z3 -d:pserver:anonymous@tuxpaint.cvs.  
sourceforge.net:/cvsroot/tuxpaint co -P  
tuxpaint  
to download the code.
```

Similarly, as an example to check out the Spamassassin plugin source for apache web server you can use the following command

```
svn checkout http://svn.apache.org/repos/  
asf/spamassassin/trunk spamassassi
```

3.3 Compiling from source for Linux and Windows

For Windows users, Cygwin provides native integration of Windows-based applications, data, and other system resources with applications, software tools, and data of the Unix-like environment. It consists of two primary parts, one is A DLL (`cygwin1.dll`) which acts as a Linux API emulation layer providing substantial Linux API functionality. The other is a collection of tools that give a Linux look and feel.



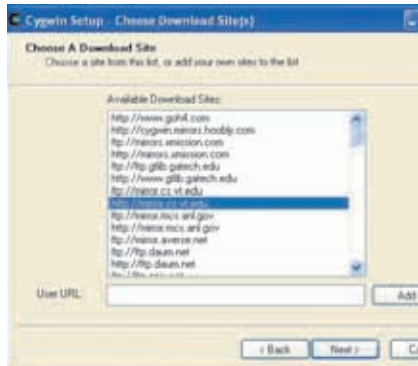
To install Cygwin on your local machine, go to www.cygwin.com/setup.exe and install the setup file on your computer.

Once you execute the setup, choose a mirror to download the packages. These are highlighted in the screenshots.

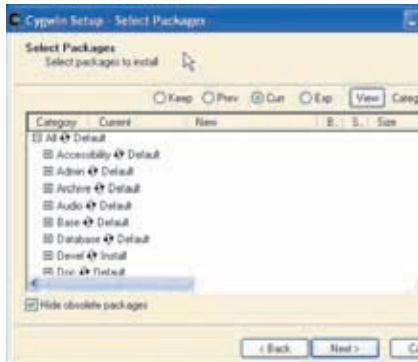
Now select the additional packages to install except the base system. we have selected “Install” next to the “Devel” category by clicking on it once, instead of the “Default” (Read not install) option. This category contains all the packages and development libraries you will need to compile your own source code.

Once you have a installed copy of cygwin, you can follow all the steps in this tutorial in the cygwin shell, for which there will be a shortcut both on the desktop and in the start menu. The steps if executed either in the cygwin shell or in the Linux console are common for both the Linux and Windows platform.

Even once you have installed cygwin, you can always open the shell and type “mount” command to see which directories point to where physically on



Select the mirror you wish to download from



Select the packages you wish to download

It should be noted though that Cygwin is *not* a way to run native Linux apps on Windows. You have to rebuild your application from source if you want it to run on Windows

your hard disk. Also, if you ever want to install/uninstall more packages, just run the setup again keeping the install location same and you can modify your selection any number of times.

Proceeding slowly about it, let us first compile a simple “Hello World” application written in C++.

First, open your favourite text editor and type in the following statements exactly and save the file as “filename.cpp”

```
#include<iostream>
int main()
{
    std::
    cout<<"FastTrack to
    OpenSource";
    return 0;
}
```

Now open your shell (either the native Linux shell or the cygwin command prompt) and by using the `cd` command change to the directory where you have stored your “filename.cpp”

Issue the following command

```
g++ filename.cpp
```

Now issue

```
./a.out
```

You have just written, compiled and executed your first application.

As the applications get more features and increase in complexity, they start having more than one source file and various other libraries that need to be packaged together for successful compilation and installation. For this very reason most open source distributions use the “GNU build system” or



For Linux users, the commands mentioned below in this tutorial need to be entered in the console, which can be opened in any of the following ways

1) Press

```
[Alt]+[Ctrl]+[F2]/
[F3]/[F4]/[F5]/[F6]
```

2) If you are in KDE, type Console in your application environment

3) If you are in Gnome, you can use the “Gnome-Terminal” application

So now we are ready to compile our first program. If you have installed cygwin with all the devel packages as mentioned above you are ready to go. For Linux users most distributions will come by default with the basic build tools, but if not you can install the build-essential package from your package manager or via command line. For example in debian based distributions, the following should do the trick

```
sudo apt-get install
build-essentials
```

as they are also called “Autotools”. Consider them as tools that make installing from source easier and feasible across system with different operating systems and hardware.

You will have to first download the source-code tree folder by one of the methods mentioned above.

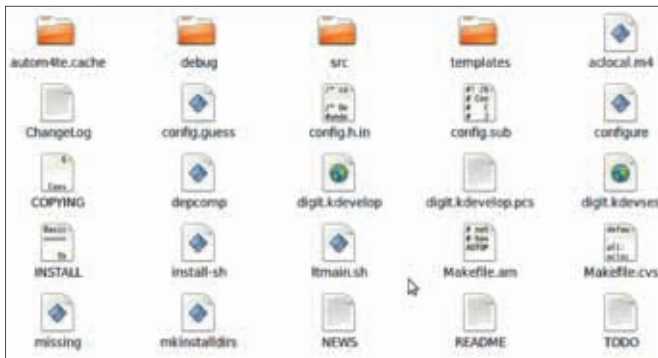
Once you have done that you will have to decompress the archive, either by using your preferred GUI archive manager or by issuing the following command in the shell

```
tar xvzf package.tar.gz or tar xvjf package.tar.bz2
```

This will extract the contents of the archive into a folder by the name of “Package”

Now let us first examine the folder in a little detail before we actually go and compile the code.

If you open the folder, you will see further folders and files in it as shown in the screenshot. The screenshot is of a “Hello World” program we created which has been packaged with the



The folder "src" and file "configure" are vital

GNU build tools.

Out of all the files above, one folder and one file is of primary importance to you - the folder called “src” and the file called **configure**. **src** contains all the source code files and



Makefile.am and makefile.in contain information for the make command

the makefiles. You can see this in the following screenshot.

Makefiles are instruction for the `make` program, which is nothing but a utility for automatically building executable programs and libraries from source code. The makefiles actually specify how to derive the target program from each of its dependencies.

In simple terms, the makefile associated with the “hello world” source code written in C++ instructs the “make” program to call the relevant compiler (g++) and that in order to compile this software, it needs the “Abra” dependency located in the “Ca Dabra” location.

Now the obvious question arises that how does the developer of the software will know where is “Ca Dabra” location on each computer. So there must be a method to dynamically generate the makefiles by examining the system and ensuring all relevant dependencies and compilers are installed and at what location. Well that is the function of the “Configure” Script above.

The configure script, when executed checks for environmental variables, install locations, necessary dependencies, tools and other stuff and if everything that is needed is there, passes on this information to the makefiles, which in turn tell the make program what to do.

The other file, “digit.cpp” in the “src” folder simply contains

```
21
22
23
24     #include <iostream>
25
26     using namespace std;
27
28     int main(int argc, char *argv[])
29     {
30         cout << "Hello Readers of Digit!" << endl;
31
32         return EXIT_SUCCESS;
33     }
```

Some of the code in the digit.cpp file

the actual source, which is as follows.

As you can see you can easily modify this source code file to suit your requirements. Similarly if you know the language in which the software that you are compiling is written, you can

easily customise it to suit your own needs.

Now that we have a decent understanding of the directory structure and what various files and tool do, let us start compiling. We have already explained above unpacked our compressed source archive. Now by using the `cd` package command change directory to the folder containing the uncompressed source.

Now type in the following command

```
./configure
```

This will execute the configure script, the `./` preceding the configure is to tell the shell that configure is not a globally available programme, but an executable file present in this folder.

The configure script as detailed above would have now produced makefiles that are like set of instructions for our “make” program.

Now it is the time to say the magic word, which is

```
Make
```

Again after expression of some serious love by means of lots of fast scrolling text on your screen, it should exit successfully.

At this point, you have compiled the source code successfully. But still the software is not installed on your system, for that you have to issue the following command

```
make install
```

Now if everything has gone fine, you have successfully installed the program. You can now execute it by simply typing the `package_name` in the shell now. In the case above, it would simply be `digit` which will nicely display the following message

```
Hello Readers of Digit!
```

To end this section, we would like to mention as to how to Uninstall the program if you ever need to, simple. Just change to the directory of your uncompressed source code and type `make uninstall` in the shell. If you haven't deleted your

Don't worry

The configure script will print gazillion lines of code on your screen, but don't worry it is normal, this is how GNU-built tools express their love. The important thing to note is what it says at the end, if it exited successfully then obviously there is no problem and we can move on the next step. But if it complains about some missing library, then just open your package-manager and installed the aforementioned library-dev package.

makefile which was created when you executed the `configure` script while installing the software, you will be able to remove any software in similar fashion that you had installed yourself by compiling from source. In case for some reasons you have deleted the makefile, you will have to remove the program files manually.

If you know where the program was installed, you'll have to manually delete the installed files. This is one of the disadvantages of building each software from source individually, instead of using a package manager.

3.4 Advanced compile options

Now if you feel slightly comfortable with the shell and the process of compiling software from source code, here are few tips and tweaks to optimise the compilation process.

One compiler flag to watch out for in the Makefile is `-g` (as in `gcc -g`). The GNU programs often have this flag, which instructs the compiler to add bulky debugging code to the executable. This is needed if you plan to use a debugger on the program. If you are reading this, then there is an over 99 per cent chance that you will never need to use a debugger on that code. So you can safely remove it by editing the makefile manually.

Some other configure options to look for are `configure --help`. This will produce a verbose output of guess what? Help!

The next configure options is `Configure --prefix=dir`. This will instruct Configure to install programs and files under directory `dir`. This can be very helpful if you want to keep your installed programs which you have compiled from source in a separate partition than the ones installed by your package manager.

For instance, to install “digit” software from the example above in the `/usr/local` directory (away from other software and easier to maintain), use `--prefix=/usr/local`.

If you really know what you are doing, or have done what you are doing a million times before, you can use `Configure --quiet` command. This suppresses the check messages.

Now, if you really want to extract the last bit of juice and are ready to get your hands dirty in the process, you can set the `CFLAGS` (for C code) and `CXXFLAGS` (for C++) environment

variables before you run the configure script.

Consider these commands:

```
export CFLAGS="-O3 -march=core2"
```

```
export CXXFLAGS="-O3 -march=core2"
```

Here we're setting up the aforementioned environment variables with optimisations suitable for the Intel Core 2 CPU architecture. But if you are a AMD fanboy and have a AMD64 chip you can use

```
export CFLAGS="-O3 -march=athlon64"
```

```
export CXXFLAGS="-O3 -march=athlon64"
```

Some other march codes are pentium3, pentium4, athlon-xp.

Now the -O3 parameter at the start tells the compiler to do the third level of optimisation. By default, generally for favour of slightly smaller size and easier debugging, most binary builds are set at -O2.

The list of possible optimisation tweaks are endless, open the makefile, try and go through each option and see what you can tweak, as always remember that "Google" is your best friend.

3.5 Reporting bugs and getting help

We have only looked till now for the utopian case where everything works as expected, but there are situations when things don't end so smoothly. In such cases, user and developer forums along with IRC channels are the best way to directly ask your question and in most cases receive answers.

If, after all searching, you are still stuck, feel free to ask your doubt. There is an expected and accepted format to do so. For example a post like the following is exactly how you shouldn't do it.

"Hey dude, my software isn't compiling, can you help???"

Here is a list of things wrong with the above:

Multiple question marks. We're not kidding, most experienced users who are you best bet have an extreme irritation to multiple question marks or exclamation marks.

It provides no details. You should provide a level of detail, via which the problem can be replicated. It is strongly advised to include the following details:

Exact name and version of your operating system, processor

details, the version of the software you are compiling, the version and name of your compiler and the exact error message that you have received; else, it doesn't reflect that you have done your own research and have put in some effort to solve it.

On the contrary, below is a good example of post requesting help.

"Hello,

I am trying to compile xyz, which I sourced from the svn repository on 1st April, 1832. When I compile on my Pentium-4 machine running Ubuntu 9.04, Kernel 2.7.1.1, I get the following error message:

```
#exact error
message, copy-
pasted from the
terminal##
```

I tried searching the forum archive and found a related solution "Abc" on one of the thread but it didn't help.

Any help on the above would be greatly appreciated.
Your name"

Consider a Source code file called "Digit.cpp" which has the following content:

```
#include<iostream>

int main()
```



Before you shoot a question on the forum/IRC channel here is a checklist:
Search The Archives: Most often than not, somebody would have stumbled upon the exact same spot as you have, and would have sought help on the forum. Every forum has a search option, use it. Not only is this a faster way but a more respected way. You will be scorned at if you ask a question which has previously been answered before on the forum.

Google is your friend: Your problem may not be as uncommon as you think, it is very likely that the answer to your question will be provided in the first page of a search engine query. If you fail to do this and ask on the forum you might be either totally ignored or meet with a "STFW" reply, which stands for Search the Frakkin Web. It should be noted that if you get a STWF reply you shouldn't be disappointed, but actually be grateful since you weren't totally ignored. Hail the hacker culture!

RTFM: This is another cousin of the "STFW" reply. It stands for "Read the Frakkin Manual", normally as explained in the Directory structure section, there will be an included "Readme.Txt" which shall have the answer to your query.

Grammar: Obvious as it may sound, bad grammar and spelling mistakes are a sure shot way to ensure that you will not get any replies.

Tip

Assuming that you have requested for help and after a discussion with some community expert, figured that the problem is not at your end but with the code itself, then you can file a bug report to inform the developers about the same, but as it is about posting a question there are certain conventions to be followed for the benefit of everyone.

Use the most updated version:

Builds change daily, and it is a pain to find, understand and communicate something which had already been corrected in the latest build.

Exact Reproduction.

You should provide enough details in it for the developer to exactly reproduce your scenario, otherwise your bug report is useless

Check the Known Issues list

Needs no explanation, you don't need to discover America all over again.

Screenshots:

Words can be ambiguous, screenshot(s) can go a long way in communicating what you want exactly, haven't we all heard "a picture is worth a thousand words".

Every Bug for itself:

Do NOT ever file more than one bug in one bug report, file multiple bug reports for each bug. This makes dealing with them and tracking them practical.

Contribution to the code

Once you feel ninja enough and think can fix a bug yourself, there is only one more thing that remains after that, which is to share it with rest of the world. For this we use the brother and sister combination of "Diff and Patch" tools. We shall explain them below with a simple example.

```
{
    std::cout<<"This is the correct
Speling";
    return 0;
}
```

Now the above code is readily compiled, but there is a spelling error in the word `spelling` and such errors often skip the eye of the developer. So now you create a file called `Digit1.cpp` containing the corrected contents as follows

```
#include<iostream>

int main()

{
    std::cout<<"This is the correct
```

```
Spelling";
    return 0;
}
```

Now instead of submitted simply the corrected file and letting the developers guess the difference (think 3000 line file), you create a “patch” file, via the following command.

```
diff digit.cpp digit1.cpp > patchfile.patch
```

diff is the name of the command.

digit.cpp is the incorrect one file and digit1.cpp is your corrected file.

> is the output redirector, which means that you are telling the computer to store the output of the “diff digit.cpp digit1.cpp” command in a file called “patchfile.patch” instead of displaying it on your monitor. “patchfile.patch” is your patch file as the name suggest.

You can now submit this file to the developers via the official mailing list which can be found on their project web site.

The developer now looking at the contents of the “patchfile.patch” can easily make out that there was an error in the fourth line and what was the error:

```
4c4
<  std::cout<<"This is the correct
Spelling";
---
>  std::cout<<"This is the correct
Spelling";
```

They can then without having to manually correct the code, issue the following command to produce and updated version

```
patch digit.cpp -i patchfile.patch -o digit_
corrected.cpp
```

This way, as explained above, in the version control system they can maintain a continuous log of what changes are being made and by whom, so that you get appropriate credit for your efforts, and if the developer find you ninja enough they might give you write access to their svn/cvs repositories where you can directly commit the changes.

Creating your Own Linux Distro

Since the first Linux distribution was released way back in 1992 (MCC Interim Linux by Manchester University's Owen Le Blanc), there have been hundreds of ways to get the world's favourite free software operating system on to a computer. The diversity of alternatives reflects the diversity in the development community, with distros split along technical, functional, linguistic and even ideological lines. There have been large distros, tiny ones, bleeding edge and rock-solid stable distros, easy for the newbie to install, or downright impenetrable to the uninitiated.

So with all the variety that's already out there, why would anyone want to create their own distro? After working with our first Linux distro years ago, we'd discover there were issues we definitely would've liked to see improved upon. We tried a number of alternative distributions to address these issues, yet each has its pros and cons.

You would use your current Linux system to develop your own customised system. This "perfect" Linux system would then have the strengths of various systems without their associated weaknesses. In the beginning,

the idea was rather daunting, but we remained committed to the idea that a system could be built that would conform to my needs and desires rather than to a standard that just did not fit what we were looking for. Anyway, creating something, from noodles to a complex web site is rewarding and satisfying – especially if you love the end result. The feeling of being in control, trying out new ideas and then seeing your goals realised can't be beaten. This is one of the joys of computing, especially programming, with the open source world a constant generator of code, concepts and communities upon which to build. And creating your own customised Linux distro, however difficult this may seem initially, is one of the best ways to combine

We believe if you want absolute satisfaction from your Linux system, you should build your own

creativity with technical learning.

We're used to thinking of Linux distributions being set in stone. They're either KDE or Gnome, use a certain kernel and bundle certain applications. But this doesn't have to be the case. You find yourself making the same adjustments each time you do a fresh install.

Making your own customised Linux distro is simpler than you might think – and rewarding too. Don your overalls and hard hat, and we'll show you how to start building your ideal distro. There are three basic ways to roll

yours, depending on the scope of what you want to accomplish and the level of technical expertise you have to bring to the project. The first, easiest, and possibly the most immediately useful to most people, is remastering a live CD. The second, and possibly the most flexible being respinning an already installed and customised system into a distro. And the third, and definitely the most hardcore and closest to 'Creating your own Linux distro' is LinuxFromScratch.

4.1 Remastering a live CD

Now, although it is possible to create a distro from scratch (see the Linux From Scratch section at the end if you want to try), it's generally better to start with a base distro. Apart from getting a head start, there are other reasons to hang on the coat-tails of another distro, and probably the biggest is package

Why create your own distro?

- You can easily build a personalised live CD by including software that does not normally come with Ubuntu.
- You are doing a presentation of a piece of software that runs on Ubuntu, and you want to put your software on a live CD to take with you to show how it works.
- You are installing Ubuntu on multiple machines and once you install Ubuntu, you always have to install specific software and drivers to configure your Ubuntu installation.
- You want to make a small Ubuntu live CD with your specific software to take with you wherever you go. This can be useful if you are in college and often use libraries or internet cafes.
- You want to create a 'Media-centric live CD' with proprietary codecs that can play various media formats, or a live CD that concentrates on a specific purpose or functionality such as games or desktop publishing, complete with topic-related applications.

compatibility. Consider that there are thousands of free software packages considered good enough to be in Debian, and many more that aren't on that project's radar.

You don't want to be compiling those yourself, particularly since dozens of releases happen every day. Also, you get to be part of the community that has formed around the base distro. Then again, free software means never having to ask permission before you create your fork.

The rise of Ubuntu has opened up one further possibility here, which is to fork from a fork. Previously, distros that forked from Debian had to cope with the outdated software problem themselves, but that was usually worth the bother because it meant you got to take advantage of Debian's huge software repository and unparalleled community testing. But Ubuntu does all that work already – it's based on Debian (even giving easy access to all of Debian's universe and multiverse repositories), then updates the software to newer versions and patches/fixes it all appropriately. The result is that you get all the size of Debian with all the latest features of Ubuntu. From your perspective, that means you can choose whether you want to base off Ubuntu or Debian, or indeed create a blend of both.

We're going to look at the methods, tips and essential knowledge required for constructing your own personalised distro, with a focus on modifying the hugely popular Ubuntu live CD. We'll also have plenty of other tips and guides for customising any Linux flavour.

How are we going to do this?

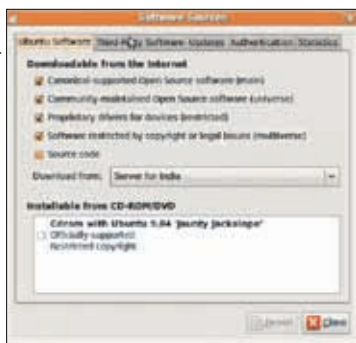
Using Reconstructor, an open source program written in Python that can create a customized Linux live CD distribution based on Ubuntu (Revisor for Fedora and SuseStudio are other alternatives). Reconstructor allows you to do extensive customisation by changing the splash screen, boot screen, sounds effect, wallpaper, programs and various other settings.

Installing Reconstructor

Reconstructor is very easy to install, with us not having to type even a few lines of code into a terminal window unless we desire to.

First of all, we have to make sure that our computer can

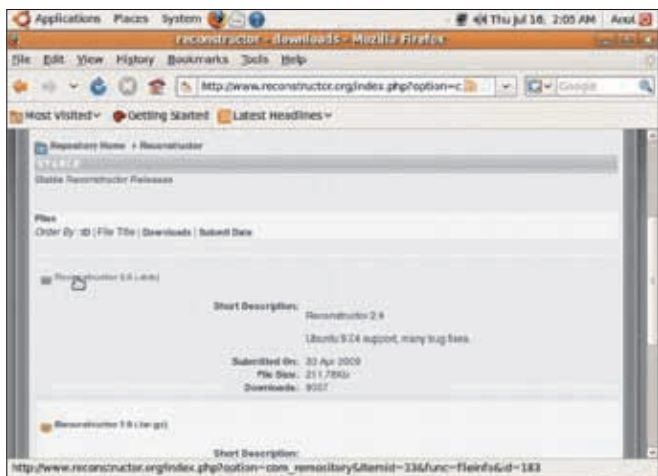
access the places on the internet where Reconstructor lives. We do this by using Synaptic Package Manager. Synaptic Package Manager is a utility bundled with Ubuntu which makes it very easy to install and uninstall software. To access it, from the Gnome menu-bar choose System > Administration > Synaptic Package Manager. Enter your system password if prompted.



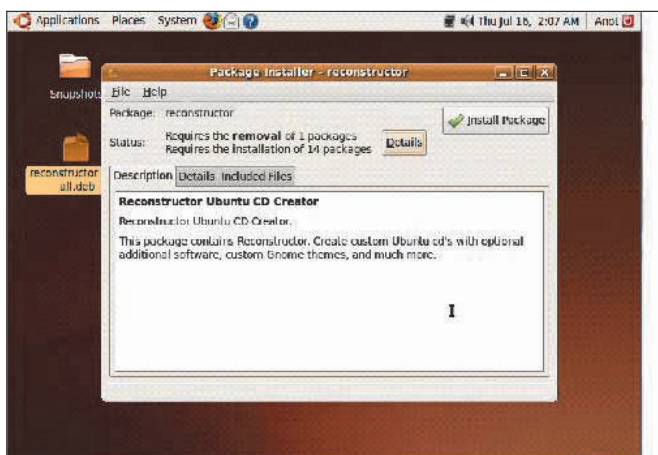
Selecting the software sources

When the Synaptic Package Manager opens, choose Settings > Repositories. In the window that opens, make sure that the following four boxes are all ticked, as showed in the image below.

Once you have ticked all four boxes, click close, and then click the Reload button in the Synaptic Package Manager toolbar.



Downloading Reconstructor for Ubuntu



Installing Reconstructor

Now, we have to download Reconstructor from the web site. In your web browser, navigate to www.reconstructor.org. From the homepage, click on Downloads > Reconstructor > Stable > Reconstructor (deb) and finally Download.

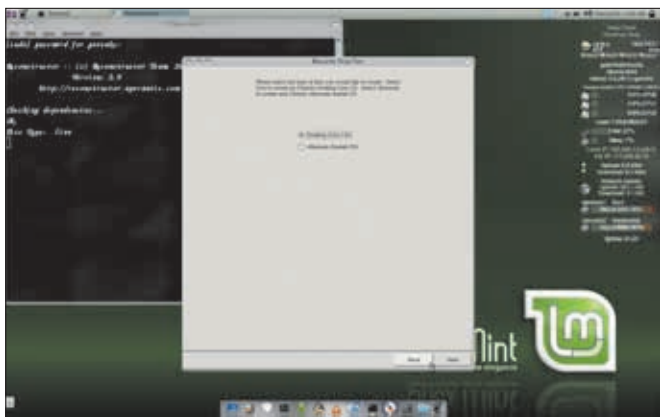
Double-click on the file you just downloaded. Next, click Install Package. Type in your password if you are asked to do so. If you follow all of the prompts, the Installer will install Reconstructor and also download any extra packages it depends on that may not already be on your system. When the install is complete, close both the windows.

And that should be it! If you have followed the steps correctly, then you should be able to access Reconstructor via the Gnome-menu bar in Applications > System Tools > Reconstructor.

Using Reconstructor

Now that we have installed Reconstructor, the fun begins. Open Reconstructor via the Gnome-menu bar in Applications > System Tools > Reconstructor, or open a shell, change directory (cd) into where it's extracted, and run `sudo python reconstructor.py`.

A terminal window might open that will ask you for your password. If it does, then enter it. Notice that Reconstructor



Selecting the kind of distribution to be created

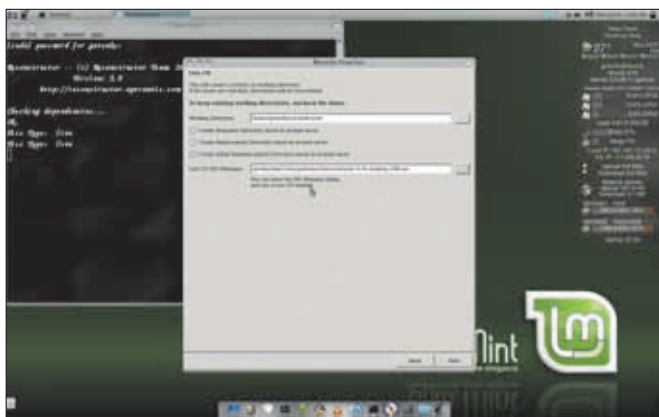
consists of two windows, the main Reconstructor window in which you use the program, and a terminal window that does all the dirty work. When you apply anything in Reconstructor, you can see it working in the terminal window, even if the main Reconstructor window seems unresponsive. Closing the terminal window will also close Reconstructor, so it's best to simply minimise it if you do not wish to see it.

Reconstructor windows

You are then given an option to edit an Ubuntu Desktop (live) CD or an alternate CD. Choose the type of disc you wish to create and click Next. In the next screen, you are given some options. Reconstructor works by copying the live CD contents to your hard drive, editing them and then compiling them to an ISO file that you can then burn to a CD.

You can then either put an Ubuntu live CD into your computer, or you can provide an Ubuntu Live CD ISO file for Reconstructor to copy from, which is much faster. If you have a live CD inserted,

If this is the first time you're using Reconstructor, or starting from a Live CD, ensure all three boxes are checked. This creates the cd-rom directory tree, the root filesystem directory tree, and the initial ramdisk tree



Setting up the working directory and location of the disc image

then leave the live CD ISO Filename box clear. If you have an Ubuntu live CD ISO file, then chose it by clicking on the “...” icon. Click Next.

If you look at the terminal window, you can see that Reconstructor is extracting the files it needs from the live CD or iso file you provided. This can take some time, and Reconstructor appears unresponsive during this time, but just let it do its thing.

Chroot Terminal

Before we talk about anything else, it is necessary to point out what we feel is the main power of Reconstructor. You should see the customisation screen with six tabs along the top, but just ignore them for a moment. If you look down at the very bottom left of the Reconstructor window, you will see a very modest Terminal icon. By clicking on it, you enter what is called a ‘Chroot’ environment. It simply stands for ‘Change Root’. What this lets you do is enter in commands as you normally do in an Ubuntu system and apply them to the live CD.

For people who haven’t heard of chroot before, in the context of Reconstructor the best way to think of it is like this. Imagine that your live CD is installed on your computer and you have just opened a terminal window. From this window, you are able to apply any commands that you want, and they

are immediately applied to the live CD. So, if you want to install some software or make some advanced customisations by using the command line, this is the place to do it.

```
apt-get clean
apt-get autoclean
```

Remove unused language packs `apt-get remove -purge language-pack*` and then reinstall English (or whatever language you want) by typing `apt-get install language-pack-en`.

Get a list of installed packages in the chroot terminal by typing `dpkg -l`. This will give you a list of all the installed packages on the live CD/DVD. From there, you can pick and choose any packages you don't need. Then `apt-get` to remove them.

Use `apt-get autoremove`. In the chroot terminal type:

```
apt-get autoremove
```

This removes packages that may have been automatically installed by something else but are no longer required.

Hint

clean & autoclean

If you have added applications to your live CD via the chroot terminal, then open up the chroot terminal and type the commands alongside

VLC

After trying out several media players, including Totem and MPlayer, we still prefer VLC for its great support for almost any kind of multimedia format out there.

```
sudo apt-get install vlc
```

Gnome Do

Gnome Do is a small application that allows you to search and do things faster and more efficiently in your Ubuntu machine. It is similar to QuickSilver in Mac and Launchy in Windows. For those who have not tried Gnome Do before, it might take some time for you to get used to it. But once you're hooked to it, there will be no turning back.

Gnome Do also comes with a dock interface so that you can use it like any other docks.

```
sudo apt-get install gnome-do
```

Eye candy

Some people like to have nice beautiful effects on their desktop

while others may just want a minimal desktop. If you belong to the former, here are some applications that you can install to beautify your desktop.

- CompizConfigSettingsManager: the configuration manager for Compiz. Inside you can find lots of interesting (and useless) desktop effects.
- Avant Windows Navigator, Cairo dock – Mac OSX style dock for your desktop
- conky, GKrellM – display system setting on your desktop.

ChrootX

ChrootX customisation works similarly to the chroot terminal except it allows you to boot into the GUI of the live CD and customise it through it's own window via Xephyr.

To launch ChrootX, click the red terminal icon at the bottom of Reconstructor's window. You'll be greeted by a dialogue that will display any desktop environments or window managers it detects on the live CD. In version 2.9 it will check for GNOME, KDE3, KDE4, XFCE4, LXDE, IceWM, Fluxbox, WindowMaker, Enlightenment, and TWM.

After an environment is chosen, it will present you with a resolution dialog. Some common resolutions have been provided. There is also an option to set the default session. What this option does, if checked, sets the default environment that the live CD will boot to. For instance, if you have KDE, GNOME, XFCE4, and LXDE installed in your working copy, and you want the final live CD to boot to LXDE, then check the box before choosing LXDE.

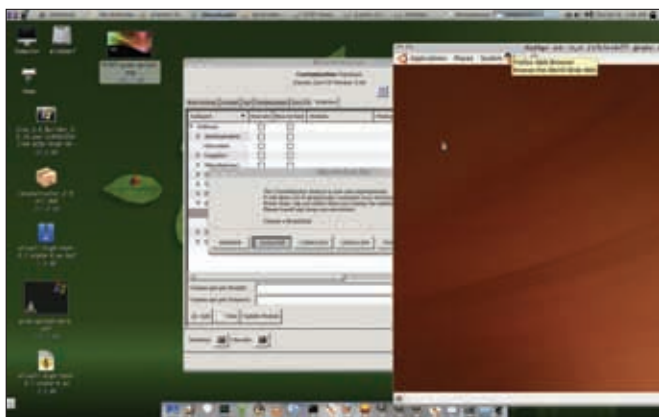
When you are done customising, log out as you normally would. Please be patient, as some environments take longer to log out from than others. Once it successfully logs out, the Xephyr window will close and you can continue with other customisations or build your live CD.

Customisation

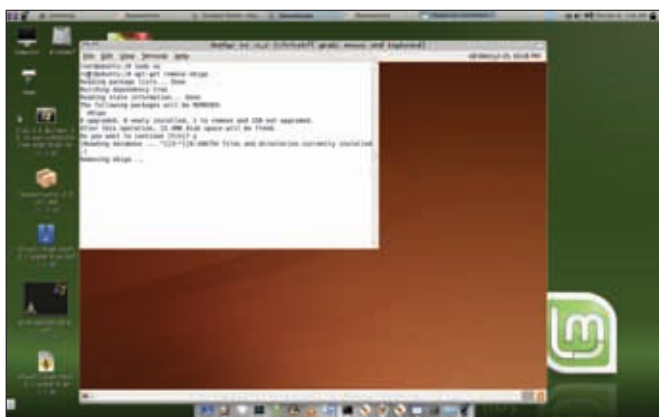
Now, let's get on with the customising!

Boot Screen

The boot screen section allows for customisation of the live CD splash screen image (the first one that is shown when choosing



Configuring resolutions



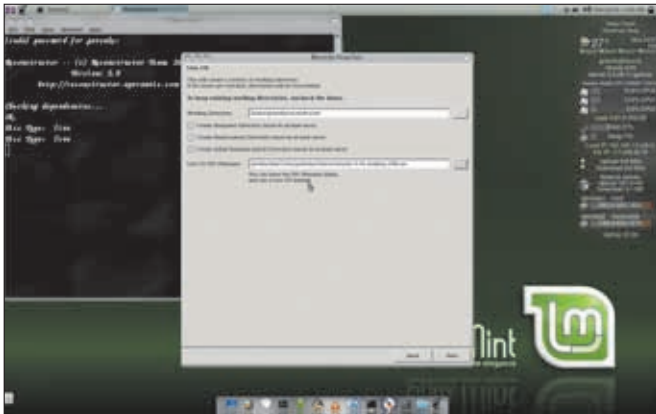
Removing the ekiga package

booting options) as well as the usplash boot screen (the image shown after selecting boot options.) To generate a usplash image, click the **Generate** button. Select the image you would like (PNG) and select where to save the library file. Reconstructor will generate the library and copy it to the correct directories.

For lots of free Gnome themes, splash screens, wallpapers, fonts, icons and more, check out www.gnome-look.org.

Gnome, Theme and Desktop

The Gnome section allows login, theme, and font customisation. The Login section allows customisation of Gnome through themes as well as options such as XDMCP, sounds, and root login. To customise the GDM theme, either select one of the default themes or click the “...” button next to the theme selection box and choose a custom theme. The custom theme can be a tar.gz or tar.bz2 package. Reconstructor will extract and install the theme into the live CD environment. You can



Customising the boot interface menu

then select the custom theme from the selection box. The same applies for the “Theme” section to select and import custom window themes and icons. The “Desktop” section allows for font and wallpaper customisation.

Apt

This section is only relevant if you plan to add software to your live CD. Some software needs Repositories added to Ubuntu before it can be installed. Reconstructor is a good example

– do you remember that when we were installing Reconstructor, we needed to go into Synaptic Package Manager and make sure

Hint

For information on creating a usplash theme, see the <https://help.ubuntu.com/community/USplashCustomizationHowto>

To make either the Official, Restricted, Universe or Multiverse repositories available, just tick the relevant boxes.

The screenshot shows the 'Properties - Group Policy' dialog box, specifically the 'Administrative (Advanced)' tab. The 'Group Policy' section is expanded, showing a list of GPOs. The 'Linked GPOs' section is also expanded, showing a list of linked GPOs. The 'Enforced GPOs' section is expanded, showing a list of enforced GPOs. The 'Enforced GPOs' section is expanded, showing a list of GPOs.

Setting up Gnome's user interface

The live CD section is used to customise the live CD environment. Options such as name, username, hostname, and password can be set here. This can be very useful for applications such as secure shell (SSH) where remote access is needed. Any value that is left blank will be the Ubuntu default value. For security reasons.

The screenshot shows the Windows Firewall console with the 'Exceptions' tab selected. The 'Exceptions' list is empty. The 'Advanced' tab is selected, showing a list of exceptions with columns for Name, Action, and Description. The list is empty.

Selecting Apt's repositories

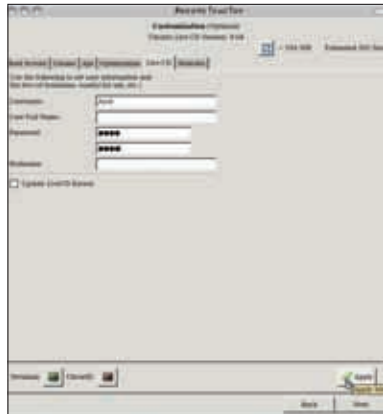
What are modules?

Modules are simply commands or scripts that can be saved as ‘.rmod’ files. Reconstructor then sees these ‘.rmod’ files as modules.

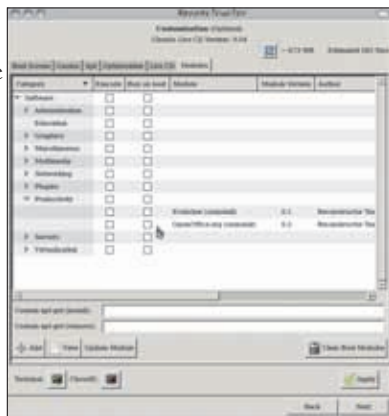
What is the point of modules?

Modules are a big time saver. For example, imagine you were going to customise ten live CDs for your local school, and you wanted to add education software to them all. Instead of using the Chroot Terminal (explained earlier) and typing your set of commands ten times, you could simply create a module with all the commands you wish to apply, and tell Reconstructor to apply that module to each of your CDs.

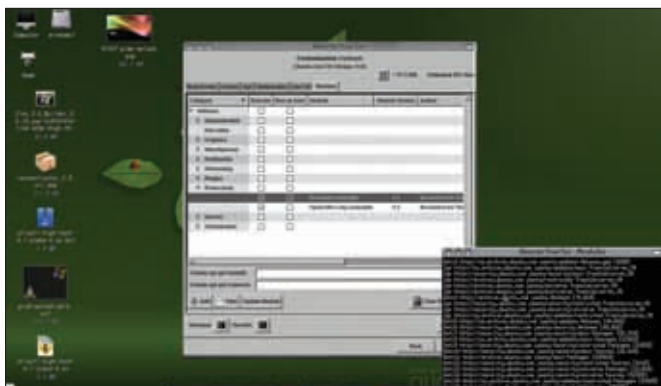
The first column consists of the names of the modules that we have installed. Reconstructor by default comes with plenty. As you can see, each module is organized by one of ten categories, from Administration to Virtualisation. Clicking the arrow beside any of the categories expands that category and shows its modules. To apply a module, you have two options. The first option is Execute. By ticking Execute, the module will be applied right away and be in effect on the live CD from boot. The second option is Run on boot. When Run on boot is ticked, the module will be processed on the startup of the live CD. This option is available due to licensing issues, when the user has to accept the license of the package(s) installed by the module (e.g. Java). Once you have ticked all the modules you wish to apply, it is important to press the Apply button,



Entering login information for the LiveCD



List of categories of modules



Adding modules to the distribution

above the Next button. A Chroot terminal window will open and apply the modules.

Writing Modules

Modules are basically shell scripts, which run inside the chrooted environment.

RMOD_ENGINE= which engine to use to process this module (basically always 1)

RMOD_CATEGORY= in which category this module belongs (normally Software)

RMOD_SUBCATEGORY= the subcategory of this module (select from a list: Administration, Education, Graphics, Miscellaneous, Multimedia, Networking, Plugins, Productivity, Servers and Virtualization)

RMOD_NAME= the name of the module

RMOD_AUTHOR= author of the module

RMOD_VERSION= version of the module, it is needed for updating the module via the “Update”-button

RMOD_DESCRIPTION= a short description of the module

RMOD_RUN_IN_CHROOT=True

RMOD_UPDATE_URL= where to find updates of this module

```
echo Running $RMOD_NAME...
```

```
apt-get update
```

```
apt-get install package name
```

```
# clean cache
apt-get clean
apt-get autoclean
echo $RMOD_NAME
```

After the configuration of the module, the packages are installed in this script by using apt-get. It is also possible to use other mechanisms, e.g. wget, and install the packages manually.

Finishing up

Make sure that all the modules you wish to apply have been ticked, and that you have pressed the 'Apply' button to apply the modules. Once you have entered all your information into Reconstructor, click Next. The final screen of Reconstructor will ask you for a name for your customised Ubuntu iso, a description and an architecture. Keep the three boxes ticked.

Ubuntu by default has a few Windows programs that you will probably never use. By ticking Remove Win32 versions of Firefox, Thunderbird, etc., you will delete these and save yourself some extra space. This will not remove the Ubuntu versions of the software. Keep the Architecture x86, unless you know what your doing.

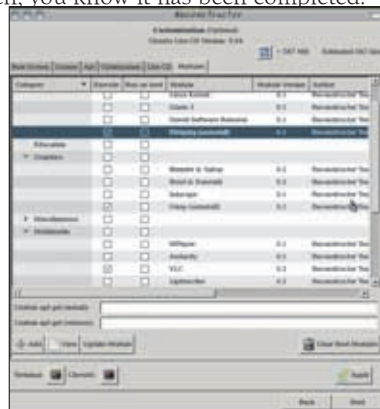
And thats it! Click next, and Reconstructor will customise your Ubuntu live disc. This may take a long time and Reconstructor again appears unresponsive, but grab a coffee, and hold on.

Once you see this screen, you know it has been completed.

Now, all you have to do is burn your new iso file to a CD or DVD. You can let Reconstructor do this by clicking on the button Burn ISO

Why would you prefer this method over others described in this guide:

You can perform all these steps in VirtualBox or VMWare without actually installing the Operating system on



Removing unnecessary programs

your computer and thus saving all the trouble of Disk Partitioning and Boot Manager Customization. It also saves valuable time and does not interfere with your master system's data/configuration apart from speeding up the whole process.

4.2 Respinning an installed and customised distro

This involves installing a given distribution, customising it, and then recompiling the distribution, modifications and all, back into an image file (typically an ISO). In the last couple of years this approach has become much easier thanks to collections of community-created tools and scripts to automate the process, so it's something that is rapidly becoming a native function for many distributions. If you're just getting your feet wet with Linux, and want to try your hand at creating a modified distribution, this is a good place to start.



Naming your distribution

The steps involved are :

1. Install all software you would like on your distro and remember to 'autoclean' after installation. This includes remastersys, so you will need to add a repo to the repository list:

```
sudo gedit /etc/apt/sources.list
```

2. Then add the following to the bottom of the file:

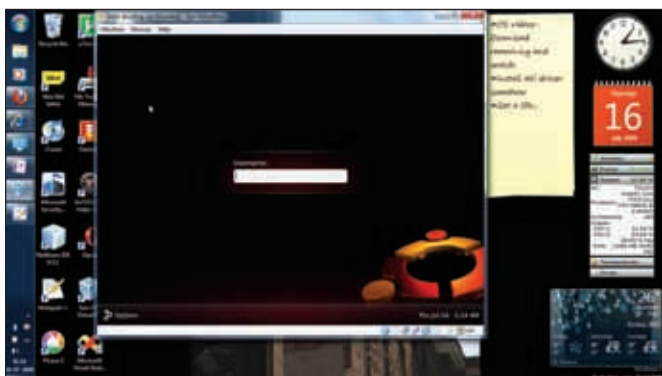
```
# Remastersys
deb http://www.remastersys.klikit-linux.com/repository
remastersys/
```

click the save button and then exit gedit.

Now type:

```
sudo apt-get update
```

to refresh your sources.list



Booting up your distribution

3. Once you're done, and all your settings are the way you would like them to be (including compiz effects setup/ AWN with launchers/ Firefox browser setup with themes and plug-ins/ menus adjusted to your liking/ wallpaper is set and icons are the way you like it/ upsplash changed/ GDM changed/ custom sound theme setup), you will need to remove all cache from your browser and documents. If you don't, these will show up in your ISO.
4. Copy the entire contents of your `/home/username` to `/etc/skel` (this is the master of all tricks) just simply drag and drop them with 'gksudo nautilus' windows open, or type `sudo cp -r /home/username/* /etc/skel`
5. Unmount or disconnect all external media or other drives in your system. Don't forget USB's, either. This will throw a monkey wrench in the works for you and you will get nothing but errors about having too much file space for a DVD.
6. Open up remastersys from `System>>Administration>>Remastersys Backup` or just hit `[ALT]+[F2]` and enter `gksudo remastersys`.
7. When the GUI display opens, just select what's shown below and then click 'OK'. The rest is cake walk, and very self explanatory.

Important

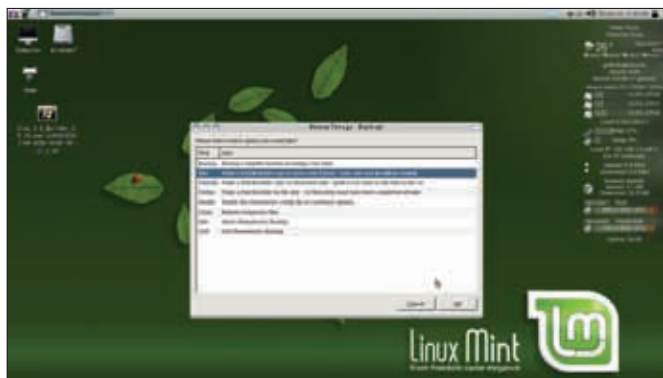
Your wallpaper will have to be transferred into `/usr/share/backgrounds` and made active from there, your icons will be in your `~/icons` or `/usr/share/icons` and so on...

When all is done, you will find your ISO file located in /home/remastersys look for the .iso file and right click it to write to disk.

4.3 Linux from scratch (LFS)

Yet another way for those who are a little more ambitious about rolling their own distribution is Linux From Scratch (www.linuxfromscratch.org).

LFS is both a distribution and an online guidebook for creating your own Linux distribution. LFS assumes that you already have a fair amount of working knowledge of Linux. At the very least, you should be able to find your way around the command line and follow directions. That said, one of the beauties of the LFS approach is that every single command you use to build the whole distribution is documented from the inside out, so you aren't just blindly following a set of instructions. The implications of everything you're doing – every



Finishing the respinning process

command, every syntax switch – are made clear to you all along. The actual creation of the new LFS Linux system is done by using an existing Linux installation, a “host,” as an environment in which to do the work.

1. Preliminaries. The first several steps involve setting up a file system, grabbing the basic set of packages needed to get things running, and setting up a few other prefatory bits like the user account you'll be using for most of the LFS work.

2. The Temporary System. The temp system consists mainly of the tool chain – a set of utilities that you build that will in turn be used to build the distribution proper, such as the GCC compiler. The tools in the tool chain are themselves compiled from source.
3. Building And Booting The System Itself. Here we actually get to begin constructing the distribution properly. As before, all this work – like creating the directories most commonly used by the system – will be done “by hand,” with details along the way about what everything is and why it’s implemented in this particular fashion. Then comes creating the boot scripts, which control the system startup process, making the system bootable, and finally starting up your newly created LFS system. The project doesn’t end there, either. There are several other entries in the LFS “family” of distribution-building projects, which you can use as the next step up. First and most likely is the appropriately named Beyond Linux From Scratch, which delves into the nitty-gritty of customising just about every aspect of your newly created Linux distribution. This is where you want to go if you either have ambitions to turn your Linux distribution into something more upscale and usable by others.

Yet another approach to the original LFS construction process is Automated Linux From Scratch, which gives you a high degree of automation for the LFS build process. The entire LFS book itself is downloaded, and the script commands in the text are extracted and run automatically. Note that this is not a substitute for reading the book; you still need to perform a certain amount of work to prepare the system, and have an understanding of the LFS build process to begin with.

Creating a Linux distribution online

One of the problems with Linux distributions found online is that they often address the requirements of everyone by putting in every single package imaginable. This makes the distribution bulky in size and also heavy on resources. For example, a distribution filled with image and video editing tools might be irrelevant for someone who wants a Linux distribution for browsing the internet and chatting.

Putting together a Linux distribution isn’t as easy as it

sounds especially for new comers. It requires patience, time and multiple runs before getting it right. Using an online service such as NimbleX Custom is simple and quick. A distribution is packaged for you and is made available for download at the end of the steps.

Using NimbleX

The custom NimbleX Live CD generator can be accessed from <http://custom.nimblex.net>. You don't need anything other than a fast internet connection to download the image and space to download it to. The site helps you create a Linux distribution using a wizard.

Welcome page

The interface is really simple. An indicator on the top of the page displays the size of the distribution at every step. Look at it when you add programs if you intend to keep the distribution in a CD or a DVD. You can select different themes for the interface using the icons on the left top of the screen.

Select the level of customisation

NimbleX offers a wide variety of customization all throughout the installation. For those who need a simple solution, there are three options to choose from – Minimal, Custom and Recommended. The Minimal option makes the most basic and minimalistic distribution available. Only the bare necessities are put into the distribution. The Custom level lets you manually choose the categories of software yourself. The Recommended mode automatically selects the most prominent applications and categories for you. You still have the option to make changes yourself once the wizard proceeds.

Using the Custom level for NimbleX

Once you select the Custom mode of creation, select the categories of software you want. This is a very important step in the creation process. If you do not mention a particular category, you won't be able to install an application from that category later in the wizard. Use the checkboxes next to the category to add the category. If you hover the mouse pointer over each of the categories, a preview of the list of applications that to be installed appears.

Selecting application categories

The categories available are office, games, network, graphics, multimedia, system, develop, libraries and drivers. The libraries and drivers are some of the most important categories present in that list. They drivers maybe necessary for the hardware on your system. The drivers add support for printers, scanners and some wireless components.

Select specific applications

When you proceed to the next step, you'll be given the option to choose specific applications from the categories you selected. The list includes a brief description of the program and also a rating for some of them. Modules can be downloaded manually as well. Moving to the next step takes you to the next category of software you selected initially. Continue the same process.

Selecting a wallpaper

The selected wallpaper will be used as a default wallpaper when you use the bootable disc. Select from the categories given. You can also upload your own wallpaper using the Upload option. Click Browse and select an image from your hard drive.

Customise sounds

The next step is set the sound preferences for the distribution. The volume tab lets you set the volume for the distribution. The KDE Sounds tab lets you choose the KDE startup and exit sounds. The last tab is the Boot audio greeting. Here, you can mention what welcome voice dialogues should be used when NimbleX boots.

Configure root account and user account

This is one of the last and most important steps. The root password is required to give you administrative permission while accessing the distribution. The login name for the root account cannot be changed but the password can. You can also set a new user account for the distribution for regular use. The next step will let you set the language for the distribution.

Finalising the distribution

Once all the steps are completed, you'll be given a preview of

the settings you've used. If you want to make changes to the settings, go back the steps. If you are fine with the settings, click on the next step. Remember that carrying out this operating on a web service is heavy, so NimbleX only allows one distribution per day, so make sure you have it right the first time.

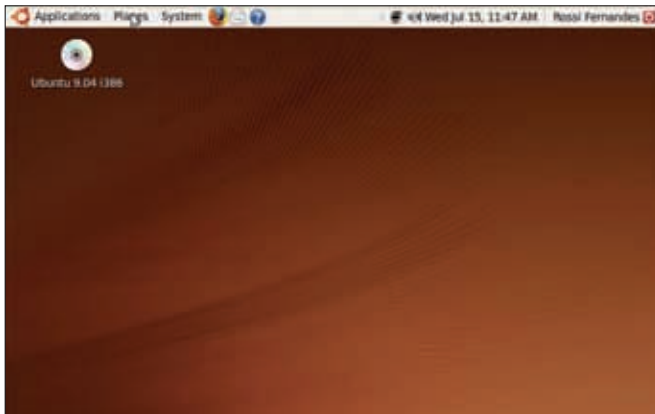
Conclusion

A lot of people make Linux sound like a wild beast of which you have to be an expert to be able to tame it. But now you have a good understanding of how to remaster a distro, we hope. For the most part, the job can be broken up into sections, as it has been performed in this guide. All of these techniques can usually be applied on almost any distro. Remember that about 90% of the distros are just remastered; what makes yours unique is what you have put into or taken out of it. Well, this guide has shown everybody what it takes to derive one's own distro from someone else's and has helped you to meet your needs. In the same sense as Ubuntu is derived from Debian, Slax is derived from Slackware, and so on.

Linux distros

5.1 Ubuntu

Linux has been around for a very long time now and being opensource, people have modified it to their needs. So, clearly there are literally hundreds of distributions and few very of these distributions have made it big. There have been distributions such as Red Hat, Mandrake, OpenSUSE that have been around for a really long time and have been extremely successful in their own way. Ubuntu is one distribution which came out in the middle of nowhere and created huge waves Today, it's easily the most well known Linux distribution



Ubuntu's desktop

around. It's well known for being easy to use and even simpler to install.

Ubuntu is ideally for people who are used to Windows and want to try out Linux. Everything has been toned down to be as simple as possible. Using it doesn't require you need to know a lot of commands. There is a similar amount of flexibility available. Usually, people who start with Ubuntu slowly move on to other distributions. The company that funds and runs Ubuntu is Canonical Ltd. There are also other projects that the company undertakes. These are Launchpad – an online

development site with collaboration features built in. There have been many smaller branches from Ubuntu itself. Kubuntu and Xubuntu are two such branches that run on KDE and XFCE, respectively. There's also Edubuntu, which is a Linux distribution that has been focussed at educational packages.

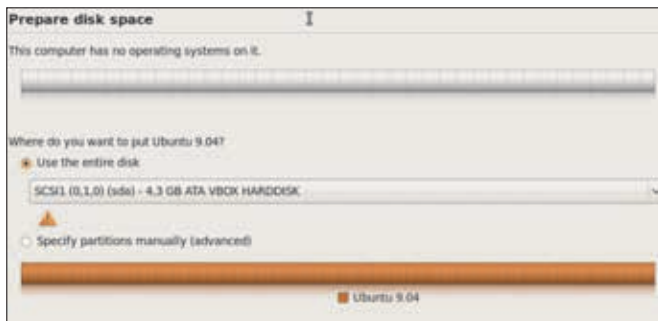
There is also a separate version of Ubuntu called LTS or Long Term Support. This is usually made for servers and there will be updates for the next few years. Even PC vendors have started shipping Ubuntu on their desktops.

5.1.1 Installation

Ubuntu is one of the easiest Linux distributions to install and is designed specifically for new comers. When you boot the disc, you are asked to select a language. You can use the [F2] key to choose a different language.

The other options available are Check disc for defects which does just that, Test memory which runs a barrage of memory tests to make sure there are no memory problems. The last option is to be chosen if you already have Ubuntu installed or you want to boot using another operating system installed on your computer. The function key [F1] to [F6] can be used to get help and information on additional parameters.

The first option - Try Ubuntu without any change to the computer runs the live CD for Ubuntu that lets you use Ubuntu without installing it on your hard drive. Install Ubuntu directly starts the installation process, but it is possible to use the first option and start the process.



Selecting a drive to install

Start by selecting a language for the Ubuntu installation and click on Forward. The next step is to setup your timezone and region. Click on your location on the map. For India, the city to select is Kolkata. The keyboard layout is next. Leave it as the default layout.

The next step is to partition the drive and is usually where people usually get confused. If you have an empty drive and want to dedicate it to the distribution, click on the drive and choose the Use the entire disk option. The installer will automatically partition the drive for you and begin the installation. If you have another operating system coexisting on the same drive, choose the Specify partitions manually (advanced) option and proceed.

There are two partitions to be created for a distribution to work. The primary partition with all the information on it is a ext3 partition and the second is a swap partition which is somewhat similar to a paging file on Windows. Click on the **New Partition** and create an ext3 partition, enter a size for this partition, and set the mount point as /. Follow the same steps to create a swap partition except here, set the Use as option to swap area. If you don't have sufficient space, you'll use a partition manager to make space on the drive not a partition.

The next step involves entering information on your user. Enter your name, a user name for the account and the password for it. Finally, give a name for your PC which will be used to

Entering personal information and login information

identify it on a network and choose if you wish to login to the account automatically.

The last step before the installation begins shows a brief of all the settings you've selected. If you need to make any changes, you should make them now. Click on Install to begin the installation. The installation may take around 10 minutes or more depending on your system configuration. At the end, the installation will need you to remove the disk from the drive and reboot. If you have Windows installed for example, you'll get a option to choose which operating system to boot into.

There is another way of installing Linux alongside Windows without partitioning the hard drive. The distribution is installed on a single file on the hard drive. Wubi is the name of this application. All you need to do is download Wubi and place in the same folder as the image file. The installation steps are pretty identical to the one using the disc.

5.1.2 Interface and Customising

Ubuntu's desktop is really simple to understand. It's similar to the Windows desktop in many ways with just the placement and locations of some of the menus changed. The panel at the bottom as it is acts like your taskbar and all running applications show up there. To the right end of the panel is the list of virtual desktops. The one on the left is active by default and the one on the right can be used if you run out of desktop space for applications. The trash bin is also present there.

Ubuntu runs on Gnome, so a lot of the distributions running on it will have the similar elements on the interface. Customizing them is almost identical. There is no single Start button on the interface instead there are several menus visible on the top. The Applications menu is the most important one and houses all the applications installed on your PC. The Places menu is used to access the different drives and network computers. The System menu has links to all the system preferences, settings and customization menus. The rest of the bar contains shortcut icons to applications like the Quicklaunch bar in Windows.

The menus and listing of applications can be modified as well. Right-click on any of the menus on the top and click on Edit menus. This window shows the menus and the submenus

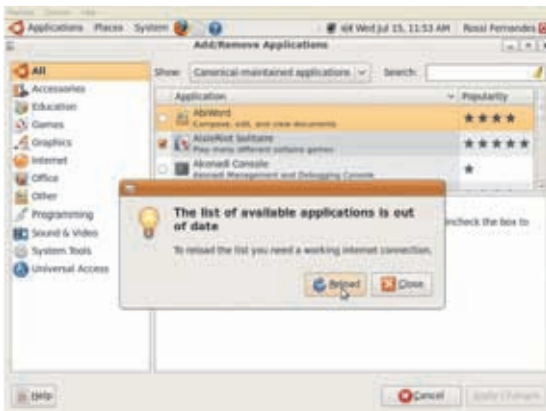
within them. You can add new items and disable the links you don't need using the checkboxes available. Adding items is done by clicking on New Item or New Menu. Enter the command to execute the application. The order of menus and items can also be changed by dragging and dropping items in the list.

The network activity icon, the clock and the current user login are displayed next to it. The right most icon lets you log in and out of the system. System rebooting and shutdown commands are also present under it.

The login window too has a theme that can be switched. To do this, right-click on the username on the right top panel, click Setup Login Screen. Click the Local tab and select a different from the list displayed. Different ways of showing the login user names are also available.

5.1.3 Installing applications

Installation of software under Linux used to be somewhat difficult with users having to use command line commands. There was the problem of dependencies as well. On Ubuntu like on many other distributions, installation is much simpler now.

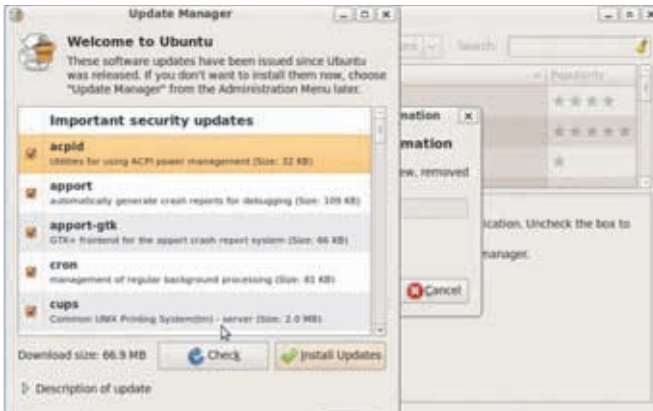


Updating the list of applications

Click on the Application menu and click on Add/Remove... Most likely, the first dialog box that appears will tell you that the list of available applications is outdated. Click on Reload to update this application list. This might take a while depending

on the speed of your internet connection. If you are looking for a specific application to install, type it in the search field on the right top. To install an application, click on the checkbox to the left of the item. Click on Apply Changes to download and install the applications. If there are dependencies that need to be fulfilled, those libraries will be downloaded and installed as well. The installed applications should appear under the Applications menu.

Uninstalling programs is equally simple. Uncheck the boxes for the applications you want uninstalled. Click on Apply Changes.



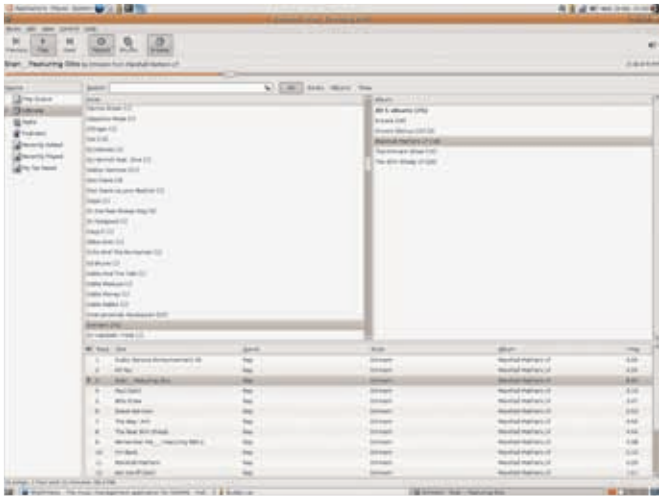
Updating the existing packages in Ubuntu

Ubuntu tries to stay updated at all times when you're connected to the internet. So, you might see popups from the Update Manager that tell you that your installation is outdated and that there are new packages available. When such a window pops up, click on the applications you want updated and click on Install Updates. This again, will take a while depending on the size of the updates and the speed of your connection.

Applications on Ubuntu are installed using the Synaptic Package Manager. The package manager is a GUI application for the APT package manager. Ubuntu also uses the DEB package format. Packages can be installed manually as well using the command `apt-get`. Open up a terminal and do a `sudo apt-get install <package_name>` and `apt-get` will download

and install the package for you.

5.1.4 Applications



Rhythmbox running on Ubuntu

The default media player that comes installed with Ubuntu is Rhythmbox Music Player. Unfortunately, it won't play every single format you throw at it – not even MP3. You'll need to install codec packs for it to work.

Like on other distributions, OpenOffice is the suite that comes preinstalled on Ubuntu. The word process, spreadsheet and presentation tool are all accessible from the Office submenu under Applications. Ubuntu doesn't come with Thunderbird as the default email client. Instead, you'll find Evolution which is a pretty impressive tool. Of course, if you want Thunderbird or any other mail client, you'll have to use the Synaptic Package manager to download and install it.

Another one of the tasks that people require are disc burning programs. There are two prominent ones in Ubuntu. The first one is an integrated component of Nautilus – the file manager and the other one is Brasero Disc Burner.

Many a times, we need to access resources on the network. Nautilus can be used. To do this, click on the network link

you can mention that too. If you don't want to keep typing this information every time you want to access the share, you can check the Bookmark checkbox so it stays in your list of Places.

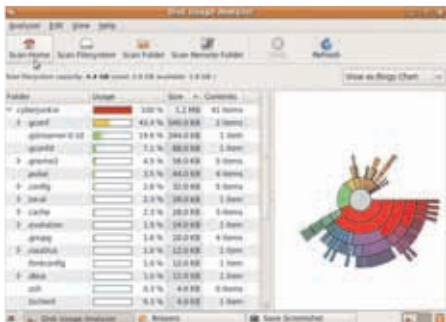
Terminal Server Client is the client used to connect to other machines remotely. It can be used to connect to Windows PCs. It's useful when you have multiple PCs on a network but you don't want to move from one place to another. Using Remote desktop gives you complete control of the PC. Obviously, you need to enable remote desktop on your Windows PC for it to work.

The Computer Janitor is a system maintenance tool that scans the distribution for programs and files that might not be in use. Those are deleted to help speed up and optimize the system.

If you've installed Ubuntu alongside another operating system, you'll be wondering how much free space you have on the partition you created. A utility called the Disk Usage Analyzer will help you with that. It can be accessed through the Applications > Accessories menu. You can also start up disk scans and check the size of a particular folder.



Brasero is a disc burning application



Disk Usage Analyser on Ubuntu

5.2 Slackware

Slackware is one of those distributions that is notorious for being difficult to work with. Slackware is generally recommended for Linux enthusiasts who want to start learning the toughest aspects of Linux. While it is true that you have to compile most packages from source, Slackware 12.2 comes with audio players, codecs and a range of applications. After the installation, you can play an MP3 file right away, which is more than what an easy distro, “Linux for human beings”, Ubuntu allows for. Slackware is one of the earliest distros of Linux to come about, and is the oldest distro still in active development. The development is done by an individual known as Patrick Volkerding, and as such the timeline is leisurely, with even the latest versions having slightly outdated software.

As Debian is preferred for its comprehensive repositories for derivative distributions, similarly, Slackware is popular because of its simplicity and reliability. Simplicity here should be read as a basic, solid and uncluttered installation, instead of ease of use. The design philosophy of Slackware is heavily inspired by the KISS principle, which stands for Keep It Simple, Stupid. Slackware is as close as any modern Linux based operating system gets to a UNIX operating system. Vector Linux, Zenwalk and Wolvix Cub are some of the more common Linux distros based on Slackware.

5.2.1 Installing Slackware

Slackware 12.0 does not have a graphical user interface for installation, but that is nothing to be afraid of. Although the installation is all in text, every step of the installation is very clearly described by Slackware, and it is not inherently more difficult than a GUI installation, perhaps simply less familiar. If you are planning to dual boot slackware, it is advisable to create about 10 GB or more of unformatted disk space using `cmd>diskmgmt` on your Windows installation. Expanding and contracting partitions is not easy during the Slackware installation, unless you use a tool such as GParted before installing Slackware.

Slip in the Slackware installation DVD into the CD tray, restart the system, and hit the [F12] key to start the installation process. The beauty of Slackware is that that you can

install the latest version of the distro even on older machines, with fewer system resources and a low configuration, and still expect it to run perfectly. Just before installation, you should see the screen as shown.

```

ISOLINUX 3.52 2007-09-25 Copyright (C) 1994-2007 H. Peter Anvin
Welcome to Slackware version 12.2 (Linux kernel 2.6.27.7)

If you need to pass extra parameters to the kernel, enter them at the prompt
below after the name of the kernel to boot (huge.s etc). NOTE: If your machine
is not at least a Pentium-Pro, you must boot and install with the huge.s
kernel, not the hugecmp.s kernel! For older machines, use "huge.s" at the
boot prompt.

In a pinch, you can boot your system from here with a command like:

boot: hugecmp.s root=/dev/hda1 rdinit= ro

In the example above, /dev/hda1 is the / Linux partition.

This prompt is just for entering extra parameters. If you don't need to enter
any parameters, hit ENTER to boot the default kernel "hugecmp.s" or press [F2]
for a listing of more kernel choices.

boot: _

```

Starting the Slackware installation

Just hit [Enter] to start the installation. The first step will be selecting a keyboard layout. For most computers, the keyboard layout is the US keyboard map, which is the default for the installer as well. Hit [Enter] again, to continue with the default layout. If you are using a non-US keyboard map, enter "1" at the prompt, and hit [Enter]. This will give you an exhaustive list of keyboard maps to choose from.

Now you will have to login as "root" to continue with the setup.

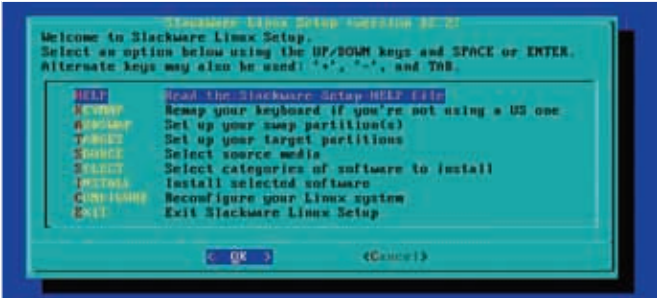
Slackware login:

Key in the words "root" and hit [Enter]. The installer screen will prompt you to re-partition your hard drive, but that can be done later on in the installation, and not necessary at this stage. You will now be on this prompt:

```
root@slackware:/#
```

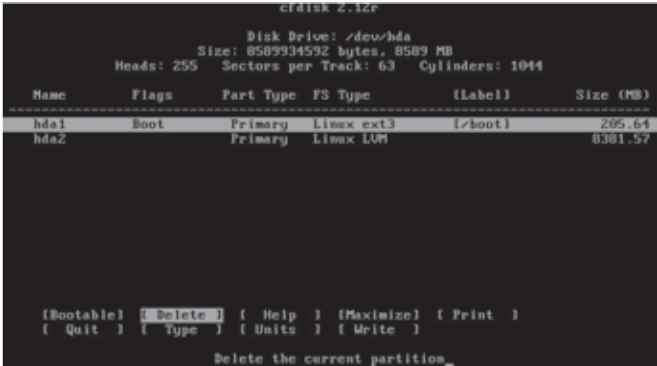
key in "setup" and hit on [Enter]. This is the screen you should see:

Now, select [OK] and continue with the installation. Apart from the partition table and the windows manager, all the



The sequence of actions for Installing Slackware.

options are pretty straight”Forward”, and don’t need any kind of tweaking by the user. CFDDisk should start up when you have to select the target partition. This is a utility that allows you to repartition the hard drive. If you are dual-booting the system, there should be an NTFS system along with an EXT3 file system. Make sure that you add ‘/’ to the primary partition where you are installing Slackware. This is a CFDDisk screen:

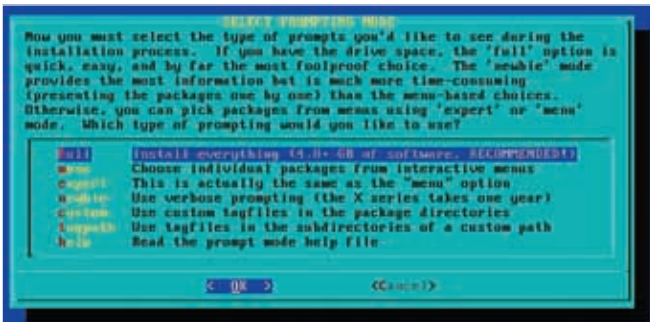


Deleting, maximizing and adding partitions

The “label” field, assigns the ‘/’ to the larger partition to make it the boot partition. This is necessary, it is the mount point for the system, and where Slackware loads up from. A Swap partition is a small amount of space used more or less like RAM. The memory from RAM is moved to the Swap space, if some other process requires the RAM memory. Using a Swap

partition is not essential, but recommended for a good UI experience. If you are stingy about space, don't go for the swap partition. If you have around 500 MB to spare, allocate it as the swap space.

You will then be prompted to select the source media. If you have burnt a DVD, the installer will detect the media on its own. You can alternatively unpack the DVD image to a USB stick or external storage, and direct the installer to this media. The installer allows you to detect the unpacked ISO from a network location, or even an FTP or HTTP server. Slackware now prompts you to select the packages you want to install.



Choosing the packages to install.

The most convenient selection on this menu is the “full” installation, which goes ahead and installs everything provided in the Slackware image. The “menu” option allows you to select individual packages that you want, but it is easier to get rid of packages you don't need later on. Selecting the packages individually takes some time, and is only for really advanced users.

You will be prompted to select a windows manager for Slackware. Slackware gives a range of options, KDE, Blackbox, XFCE, twm and fwm2. The KDE version on Slackware 12.2 is KDE 3.5, which is not the latest version of KDE. XFCE is a light weight windows manager, and for the geeks, twm is really lightweight and very basic. After the installation, the windows manager can be changed using the “xwmconfig” command in the command line.

That's it, your Slackware installation should begin now,

take a break, have a cup of coffee and come back.

Now one of the things about Slackware is that the GUI does not start immediately on startup. After Slackware starts up, you will have to login, using the username and password. At the prompt, key in “startx” to start the X windows manager. This is whatever windows manager you chose during the installation process.



First run after installation

5.2.2 Default programs and installing programs

If KDE is selected as the Windows manager, KOffice is installed as the default office suite. Slackware is geared towards programmers in the sense that a bunch of coding and debugging tools are also entered. Like apt-get in Ubuntu, the command for getting applications for Slackware is slapt-get. There is unfortunately no default program for resolving dependencies automatically, and users will have to initially wade through dependency hell for some programs. However, because of the wide range of programs installed by default, and the coding frameworks, the lack of dependencies will be few and far between, which will reduce drastically on use.

Slackware uses .tgz packages for installation. If you are downloading a software for Slackware, and don't want to compile the software, look for the .tgz package on the software website. If one isn't there, chances are, there is an unofficial build. Scout around in the forums and on Google. Once you get hold of a .tgz package, use the

```
#installpkg (packagename)
```

Command. Another interesting tool that Slackware provides, is support for .rpm packages. There is just another step in between. As .rpm packages are easier to come by, this is pretty convenient. Use the following command.

```
#rpm2tgz (packagename)
```

If the package name is `software.rpm`, the package will now be `software.tgz`. Now use `installpkg` to install the software. Who said Slackware was difficult? You don't have to compile everything.

5.3 OpenSUSE

OpenSUSE is another one of those distros with a long history, the first version having come out in 1994. SUSE was originally meant to be a commercial, open source, distro. In terms of usage, SUSE for commercial use comes in second only to Red Hat. Novell took over the company a decade later, in 2004, maintaining and providing support to several SUSE editions. OpenSUSE is the edition which is free for public use. There are enterprise and server editions also available. OpenSUSE is well known for being one of the most actively maintained distros around, with around four OpenSUSE releases every year. The current version is OpenSUSE 11.1. There are CD and DVD versions available, the CD version having an office suite, some image editing suits, and both KDE and GNOME available as desktop environment choices, with a few windows managers thrown in for good measure.

However, setting up OpenSUSE and installing it is not terribly easy, and is one of the more frustrating experiences around. To start with, as OpenSUSE is a commercially backed distro, similar to Ubuntu, the codecs don't come in with the distro. For the more common codecs, this is not too much of an effort after installation, but for the more obscure codecs, it becomes a pain to set them up.

OpenSUSE is a great general purpose distribution, is frequently updated, and has a great community around it for any kind of support that users may want. SUSE has a large and comprehensive repository, and almost every software for Linux out there has a build for the latest SUSE version. Parts of the Novell team working on SUSE, also work on KDE and GNOME. The distribution usually has the latest versions of the desktop environments in its releases, and is a widely used distro, particularly in India. If you have a decent internet connection, OpenSUSE should pose no real problems as long as you are willing to take time to figure things out, and have the patience to work with a distro that gives very little out of

the box. For power users, with limited space on their laptops, this is a very good choice, as OpenSUSE does not clutter the installation with unnecessary packages. There are a wide range of .iso files available on the SUSE website for OpenSUSE. You can download GNOME or KDE only ISOs from the OpenSUSE website, as well as a MiniCD version of OpenSUSE.

5.3.1 Installing OpenSUSE

Open SUSE uses YaST, which stands for Yet another Setup Tool. YaST is one of the most versatile Linux applications ever written, and one of the strong points of OpenSUSE. YaST can not only install the operating system itself, but also install the packages later on, from within the operating system. There are also some advanced options available, for example, you can go into the console while OpenSUSE is loading, mount a drive, and print screens from the installation process itself.

Place the OpenSUSE media in the tray, and restart the computer. Hit [F12] and select the CD/DVD drive as the first boot media. The OpenSUSE installer should start. The first screen allows you to Boot from Hard Disk, Install, Repair an



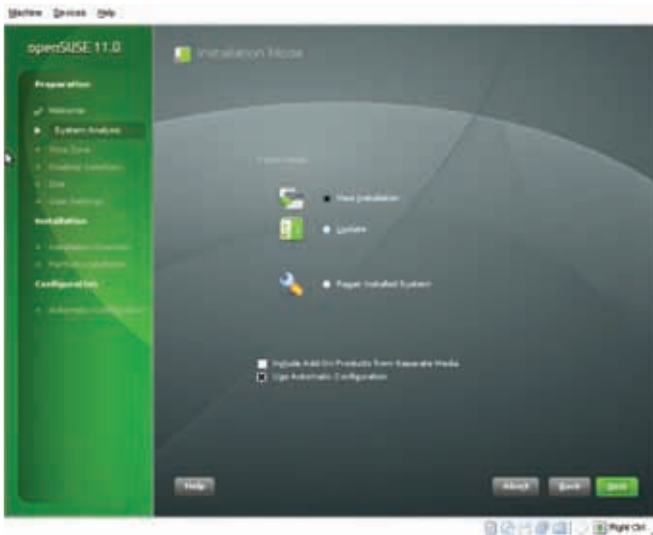
The installation welcome screen

installed SUSE system, and run some diagnostics on the host machine. Select “Installation” and hit [Enter].

If you want to change the language of the installation at this point, hit [F2] and select a language of your choice. Note that this is the language for the installer itself, and not a language for the actual installation of the software.

The installation of OpenSUSE has a very well designed GUI, and feels like a Wizard. You can see all the steps along the left side of the screen, along with an indication of the current step in the installation procedure. On the first screen, you get to choose the language of the OpenSUSE installation on your hard drive, and the default map of the keyboard. Go through the License agreement, which is very brief, and check the box that says “I Agree to the License Terms”. Then click on next.

The installation will now be in the second step of the wizard, which is “System Analysis.” The OpenSUSE installation now “probes” your computer, to find out the peripherals and the devices attached to it. This is an entirely automatic procedure, and just informs OpenSUSE of what drivers and controls to install. Once the probe is complete, an “Install



Selecting the installation mode

Mode” screen will show up. This is still the “System Analysis” step of the installation. There will be three choices given to you, “New Installation”, “Update” and “Repair installed system.” If you have been using a previous version of OpenSUSE, such as 10.0, and are using the 11.1 installation now, update is the option. Otherwise, go for “New Installation”. The “Use Automatic Configuration” checkbox is selected by default, if it is not, make sure that it is checked. For an advanced mode of installation, uncheck this box.

Next, you will be prompted to select the timezone. For Indian users, this is Kolkata. Howering over India is sufficient to select this option. Check the box that says “Hardware Clock is set to UTC” and click on “Next”.

The installation goes in the “Desktop Selection” step now, which gives you a wide selection of Desktop Environments and Windows Managers to choose from. GNOME is available, as well as two versions of KDE. Gnome works marginally faster than the latest version of KDE, and is known to crash less, but KDE is more eyecandy. Right now, you will face a dilemma that many Linux users have been facing throughout Linux history,

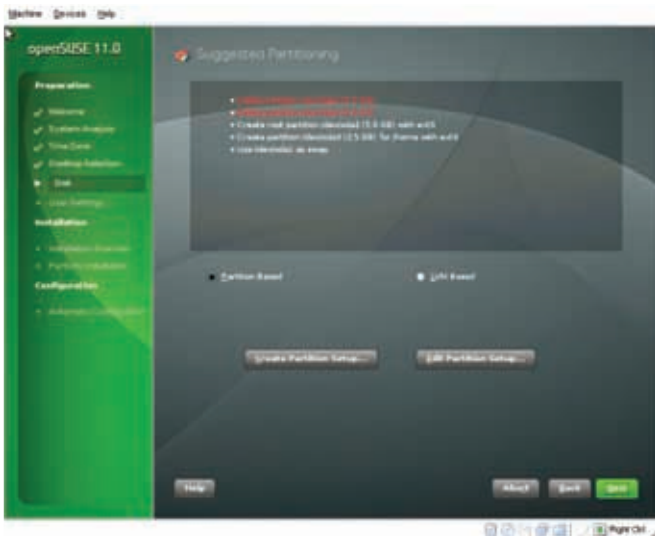


Choose a desktop environment

GNOME or KDE? We cannot settle THAT argument, so toss a coin or something.

Optionally, you can choose twm, XFCE, Fluxbox or other Windows Managers if you select the “Other” option on this screen. You can also choose not to install a desktop environment at all, if you are comfortable with the command line. Click on Next.

Now the setup will be in the Disk step, which chooses the partitions where OpenSUSE will be installed. YaST is the partition manager as well, and can resize existing partitions to make space for the OpenSUSE installation. Click on create partition setup to edit the partition table. You can either choose to use an entire disk, or the empty space in an existing disk. OpenSUSE selects the mount points automatically, and prompts for swap space. Once your done, your screen should look like this:

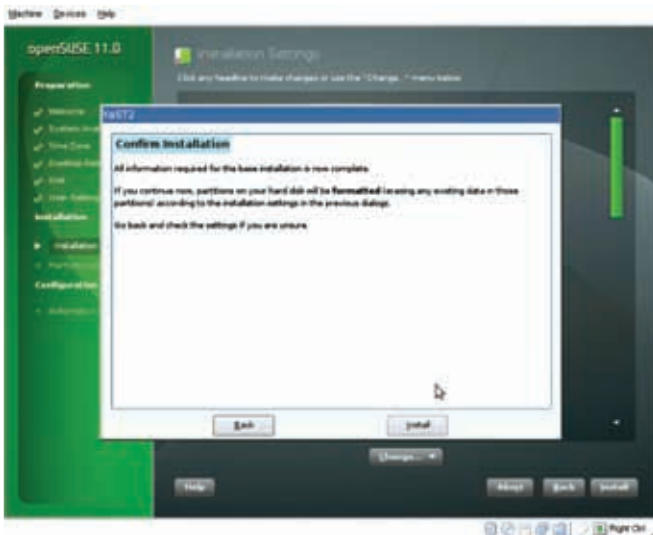


Setting up the installation target

The next step is creating the user account. Enter your full name, your username and the password twice over. Most common passwords will be prompted as being too easy to crack, so just ignore this warning if you are working on a personal

system at home. The alternative is to set a strong password containing letters in both uppercase and lowercase, as well as a few numbers thrown in. Click on “Next”.

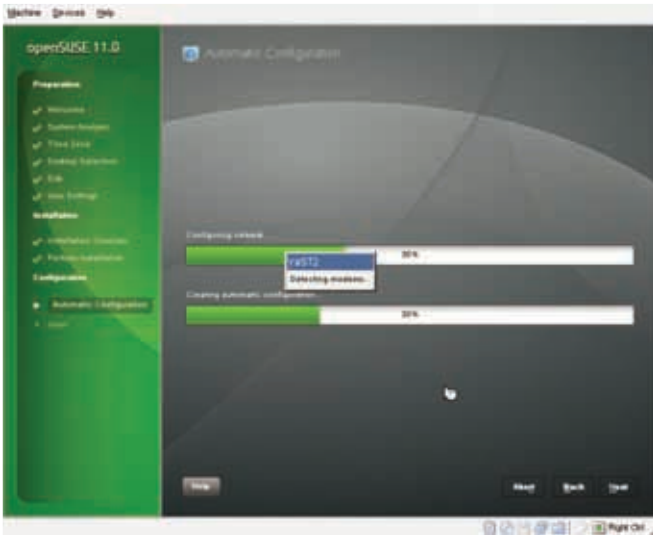
So far, the installer was in the “Preparation” mode. Now it goes into the “Installation” mode. All the options you have selected so far, including the partition table, haven’t made any changes to the system. You can click on the “Change” button at this point, and go back to any step in the installation so far, while leaving the other options unchanged. If you want to control the software installed on your machine, or are short on disk space, select software from the list that pops up when you click on “Change”, and go through the list of software, deselecting the ones that you do not want to install. This is a time consuming and laborious process. Once you are done, click on Next. You will be prompted with a confirmation. After this point, OpenSUSE will start installing the operating system on your Hard Disk, and there may be some permanent damage if the right options were not selected.



Pre-installation prompt

Now the installation process should start. This will take some time, but while OpenSUSE is installing, there is a Slide

Show that is being played that familiarizes new users with the distro, and gives some tips and tricks about how the distro works. Sit back and pay careful attention to this slideshow, as it will ease the cross over from other distros or operating systems to OpenSUSE for new users. At the end of the installation, the system reboots once. After the reboot, the installer screen is still running, now in the final step. This step is the automatic configuration of OpenSUSE, and sets up the peripherals and the devices connected.



Automatic configuration

That is it, your OpenSUSE setup is now up and running.

5.3.3 Installing applications

There are many ways to install additional applications in YaST. You can fire up YaST from the menu, and select a number of packages like the package manager in Ubuntu.

Optionally, you can also use the one-click installation method, which is something unique to OpenSUSE as far as distros go. The one-click installation method works for most sites with Linux software, or even drivers and codecs. When you navigate to the web site of the software from a OpenSUSE

installation, irrespective of the browser you are using, you should be able to see a button like this.

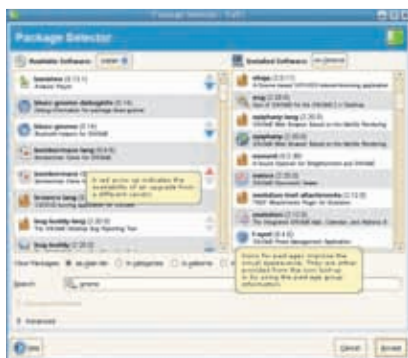
Just click on this button, and the program starts installing, downloading the packages and the like. One irritating feature is that you will be prompted to agree to saving every small file during the installation process, which considerably slows down the installation, even if you don't bother to read the prompt after the first few times.

For all it's uses, YaST falls short on one count – resolving dependencies. OpenSUSE uses ZYpp for that, which is a slightly complicated process. If

you find an obscure bit of Linux software, chances are that you will have to wade through dependency hell.

To install codecs on OpenSUSE, just to get you started off on the basic stuff like being able to play back MP3 files, head over to <http://www.fluendo.com> and “buy” the mp3 codec for zero Euros. This process is necessary because the mp3 format is proprietary, and it is illegal to distribute it freely. You will have to sign up, which is necessary only because the “dealers” have to track the “sales”. There is a one-click installation for the codecs at the end of it.

Overall, OpenSUSE is a very well built, robust and reliable distribution. It is one of the most well documented distros around, and the documentation for OpenSUSE is a valuable



Installing programs from YaST.



One Click installation button

resources, almost as much as the distro itself.

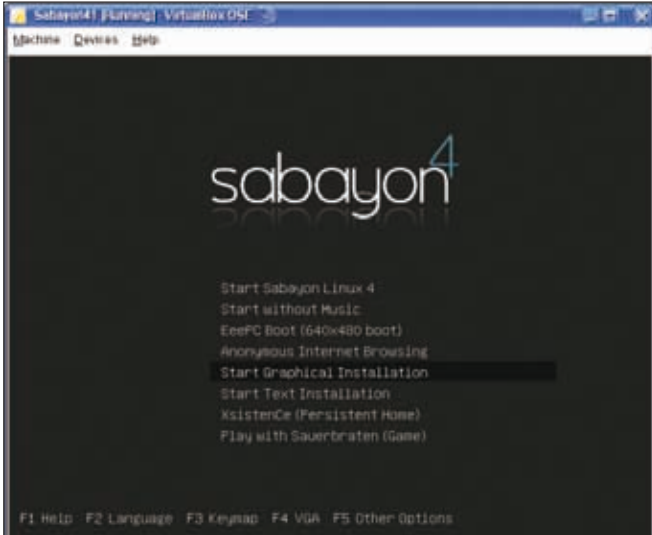
5.4 Sabayon

Sabayon a newcomer as far as distros go, but has quickly become one of the more popular distros around. Sabayon is based on Gentoo, moving away from the typical Debian/Red-Hat route. Gentoo is one of those distros that encourage you to totally customize it and build it for your hardware, streamlining it for the purpose you want to use it for. Sabayon however, moves away from the concept considerably. Sabayon is to Gentoo – a distro widely regarded as being difficult, what Vector Linux or Wolvix Cub is to Slackware, a distro that significantly eases the transition into the more difficult distros in Linux. Sabayon is one of the few distros that aim to look good and actually manage to do it without being over the top – like Linux Mint or Vector Linux. The somewhat toned down eye candy is relaxing to look at, and Sabayon is an example of one of the more effective and modest implementations of KDE. Sabayon is in active development, and there are regular releases available. More importantly, Sabayon is one of the few gaming distros out there, and you can start playing a few decent Linux games seconds after installation. The current version is Sabayon 4.0, and is pretty up to date. Sabayon releases are tied in with the development timelines of both GNOME and KDE, but there are builds with alternative windows managers and desktop environments available. Every version of Sabayon has the XBMC media centre included, and most of the codecs are included in the build, which makes it less of a hassle for even new users. If you have a great system, want a lot of eye candy, and are prepared to experiment with Linux Games, Sabayon is most certainly the distro for you.

5.4.1 Installation

The newer versions of Sabayon use the Anaconda installer, which is the same GUI installer used by Fedora. Sabayon also works as a LiveCD, if you want to have a look at the operating system before installing it. Place the Sabayon 4.0 DVD in the tray, restart the machine and the installer screen should show up. Sabayon has a very thoughtful installer, which allows you to fine-tune the installation in many ways, in the spirit of Gentoo

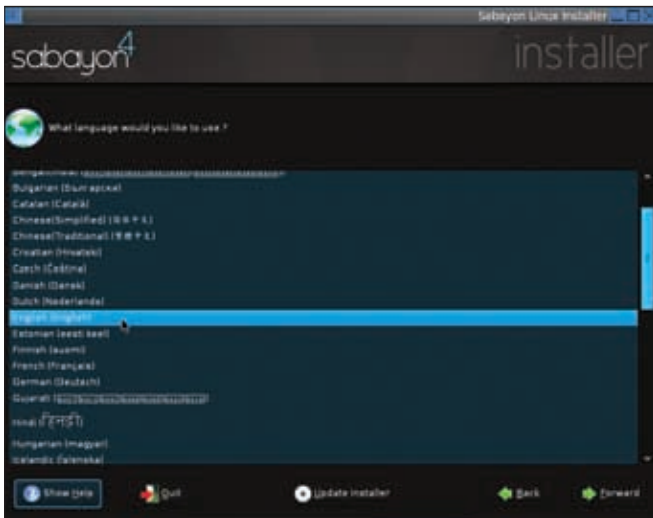
Notice the deviations from the typical Linux installation. There is a special version of the installer for EeePC, and similar systems, with a reduced display resolution. You can also start



The initial screen of Sabayon 4.0

playing Sauerbraten right away, which is a 3D first person shooter based on the Open Source cube game engine. If you are comfortable with another language, hit [F2]. You will be surprised at the number of Indian languages available. This option just changes the installer language, while not affecting the language of the actual installation, which is the language of the Operating System itself. Choose “Start Graphical Installation” to proceed. The next screen is just a welcome screen, click on “Next” after reading the message.

The next step is to select the language. This is the language of the Operating System itself, not of the installer. The menu and the applications within Sabayon will be in the language that you select in this menu. Select a language and click on “Forward”. The next screen is the keyboard map. The default for most keyboards is the U.S. English map, choose this and proceed. Even if you chose a different language for the installer, choose the U.S. English map in this screen, as irrespective of



Selecting the language in Sabayon 4.0

the language on the machine, the keyboard itself is likely to be a U.S. English based keyboard.

The next screen throws up a selection of desktop environments. KDE is the default, GNOME, XFCE, a variant of XFCE optimized for netbooks, Fluxbox and an option not to install a desktop environment at all. Note the lack of Windows Manager options on this screen. For any other distro, we would not suggest either GNOME or KDE over the other. However, if you don't have a preferences already, then for Sabayon, KDE is a better option, simply because KDE looks better, exploits good computers, and the folks at Sabayon seem to have worked just a little bit more on the KDE version.

Click on “Forward”. On this screen, you choose a whole bunch of packages to install. Sabayon offers a compromise between choosing each software in the installation individually, and choosing to install everything at one go. Instead, it offers broad categories of software, and installs everything in that category once it is selected. Check the categories you want to install, and uncheck the categories you want left out. Notice that if you select one desktop environment, the other desktop environments show up in this screen. Installing the Gnome

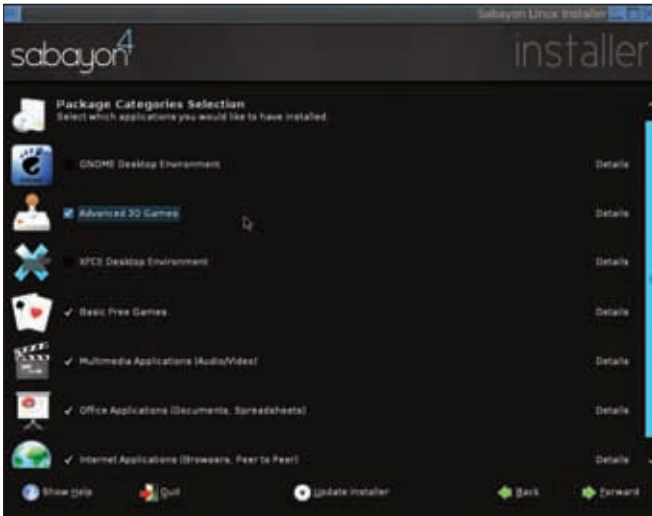


Choosing desktop environments in Sabayon 4.0

desktop environment along with KDE is recommended, as this allows you to easily install additional software for GNOME, which is pretty useful. Although there are KDE alternatives to software such as F-Stop, F-Stop is just better at what it does. Down the line, if you need some GNOME based software, then installing it at that time will be a little more difficult than installing it in this screen itself. If there are space constraints however, deselect other desktop environments. Make sure the “Advanced 3D Games” choice is checked, this will let you get a taste of some good Linux gaming as soon as you install. Click on “Forward”.

In the next screen, Sabayon goes ahead and allows you to select individual applications and packages in each category. It is not recommended that new users go through this list carefully. Just click on “Forward”, and be on your way. The next screen is for advanced users, and allows you to open or close ports depending on your usage. If you are unsure of this screen, go ahead with the default options.

The next step is setting up the partitioning. The default partitioning creates one system disk, and one swap partition. This is idea for most users, but some prefer a smaller system



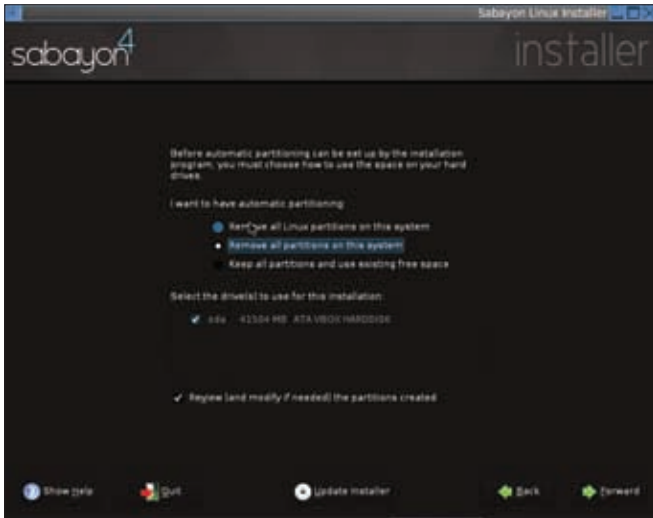
Selecting package categories to install.

partition, and a larger file partition. This really depends on your usage and what you are comfortable with. You can fire up the LVM and partition the system in a GUI. It is however, simpler to go ahead with the default layout. Sabayon is thoughtful enough to detect other operating systems on your machines, and offer to dual boot instead of removing the host OS. You don't even have to go into the



The services configurations dialogue

partitioning table for this procedure. Just select “Forward” with the default layout, and you should see this screen:

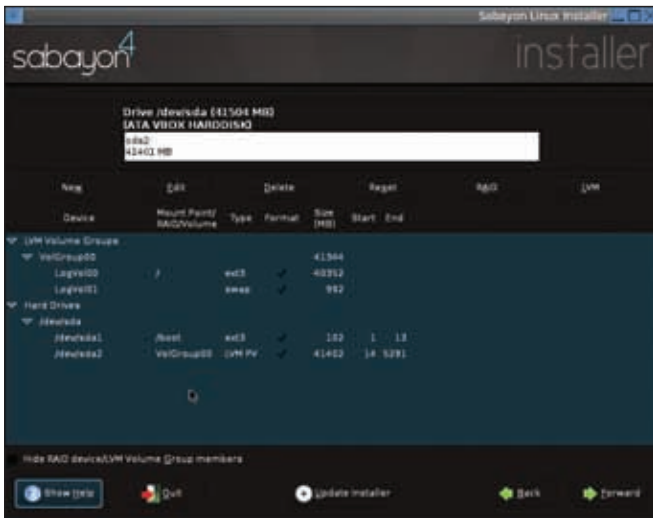


Dual Booting or replacing OS

Select “Keep all existing partitions and use free space”, which is the third option, to keep your existing operating systems, and dual boot it with Sabayon. The first option, which is “remove all Linux partitions on this system” replaces only an older Linux installation, or anything that uses a Linux filesystem. The second option, “remove all partitions on this system” replaces all the data on the hard drive with a brand new Sabayon installation, and is not recommended if you have important data stored on that hard drive. Click on “Forward”. You should be presented with the partition layout, which will look something like this:

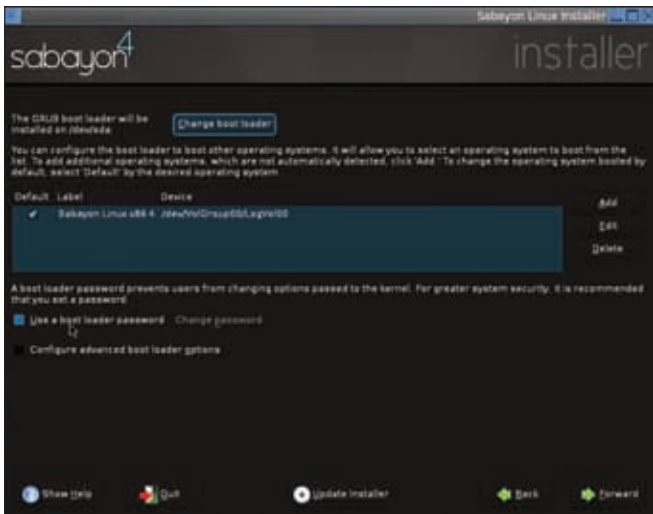
Proceed with the installation by clicking on “Forward”. Sabayon uses Grub as the bootloader, which is an interface before an Operating System starts up on the computer. This allows users to select the OS on a screen. Unfortunately, Sabayon does not offer Lilo, a better bootloader which supports graphics.

Notice the check box for a bootloader password. If you



The partition layout in Sabayon

enable this, and key in a password, users accessing your machine cannot start up the computer at all, without entering a



selecting a boot loader



Assigning user accounts and passwords.

password at the bootloader screen. Click on “Forward”. In the next screen, it is necessary to key in your internet connection details. This includes your IP address, gateways, and DNS servers. You can leave these fields blank for now, if you don’t know the details or don’t plan to use the internet on the installation. These details can be put in later as well. Click on “Forward”. Then, you will have to select the timezone, which is Kolkata for Indian users. Click on “Forward”. The next screen is for user account management. There is a root account, a superuser account, which is a user account with root access, and as many user accounts as you want. Linux was built with a network environment in mind, so the security might seem excessive, but this is the kind of access that people at AntiVirus companies with all Windows users could do with.

Once you are done filling in the details, click on “Forward”. The installation should now start. This will take some time, so wait for it. Sabayon also has a neat little slideshow showing the various features of Sabayon while you are installing it.

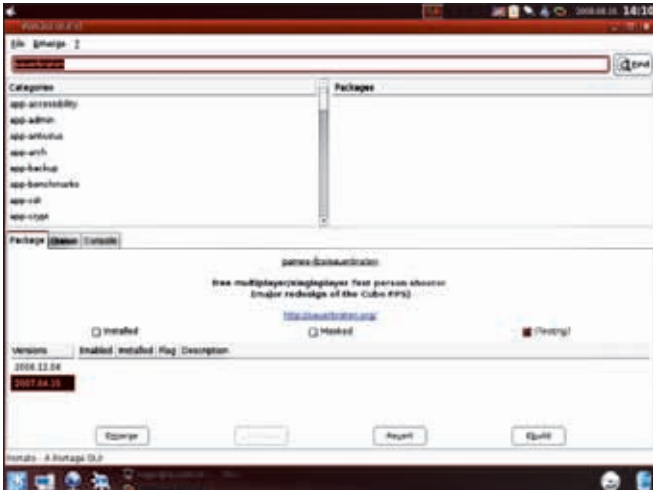
5.4.2 Installing applications

Almost everything you can think of is there out of the box in



Sabayon 4.0 in the process of being installed.

Sabayon. You will really have to work hard at finding a software that you will use regularly, that is not already there in Sabayon 4.0. Most movies and music run right away, there is a great



Using the Portato package manager.

selection of the best Linux Games available, and using it is a thrilling and refreshing experience as far as Linux distros go.

The package manager is called Portato, which is a strange little thing that is based on the Gentoo package manager, portage. Portage, in fact a great package manager compatible with a wide range of packages, as it was designed on the Gentoo distro for being compatible with BSD packages.

Search for an application in Portato, and click on Emerge. Portato resolves dependencies automatically, so there is not too much of a dependency hell here.

Sabayon is a great multipurpose distro to work and play with, and is a better replacement for a complete Windows system simply because of all the things it manages to do out of the box. In this sense, Sabayon is “easier” than Ubuntu, because of the availability of codecs, the default KDE in Sabayon looks much better than the default GNOME in Ubuntu, and you will have a highly polished distro in your hands before you know it. Another plus is the attractive XBMC media center installed on the system, which provides a great interface for movies, videos and music. If only Sabayon could support Crisis...

5.5 Debian

Debian is one of the most comprehensive Linux distros out there. Debian supports a wide range of architectures, and has a huge repository of over 25,000 software. Every one of these 25,000 software is free, and more often than not, open source. Debian has taken the vision of Linux to heart, and is developed by a massive collaborative effort. For example, just because the name “Firefox” was trademarked by the Mozilla Corporation, Debian uses a browser branded as “Iceweasel”. This also means that proprietary codecs are not included, and can be a pain for new or inexperienced users. A variation of the Linux kernel is being made, called Hurd, which is still some time away before being released. The latest version of Debian came out in 2009, and is called Debian 5.0, or “Lenny”. Debian is a distro that can be made to run on anything that has a processor of any kind. Although Debian has been around for a long time, the development is relatively slower. This does not mean that the project is any less active, just that all efforts are made to make sure that every release is reliable and stable. While other distros

are already in double digit versions, Debian has just reached a version number of 5. Despite the focus on reliability, an approach that both Slackware and Gentoo used, Debian still supports one of the largest software repositories around. In fact, there are unofficial repositories, for installing software that does not entirely fit into Debian's strict requirements of free and open source software.

Due to the immense size of the repositories, a lot of distros are derived from Debian. Any distro that uses the .deb packages, is a derivative of Debian. Knoppix and Ubuntu are the major ones, and both these distros are widely used and well known, even having a whole bunch of derivative distros of their own.

The entire distribution can be downloaded as 35 CDs or 5 DVDs. Since just downloading the distribution takes up a considerable amount of resources, most people who actually download and install Debian go for the net install version. This installs the basic operating system, after which users will have to install packages from the internet. The netinstall CD is just short of 700 MB, and this is in fact, not a bad idea. Most users will use just a fraction of the 25,000+ software supported, and a fair share of these software are small and easy to download and install. Debian shies away from having "versions" of the distro, and instead uses something known as "melds". Every installation of Debian grows into a "meld" over time. A "meld" is a Debian build with a selection of packages suitable for a particular purpose, as in network administration, education or gaming.

5.5.1 Installing Debian

There are two basic approaches to install Debian, the difference is very superficial. Place the disc in the tray, and restart the machine. This is the screen that you should see:

The options available are Install, Graphical Install, Advanced Options and Help. Choose either the Install or the Graphical Install options. The steps in both the installations is essentially the same. We will be showing steps from the graphical installation. An interesting feature in the installation for Debian, is that it allows you to take screenshots, like the OpenSUSE installer. However, you do not need to go into the command line and mount a drive to do this, there is a button on the installer itself. Next, you will have to



Debian installation start screen

choose a language and a country. Notice that Debian does not have the typical world map for choosing the Country.

The partitioning part of the installation comes up now. You can choose to dual boot Debian with another installation, but it



Choosing a country

is safer to create the free space before the installation. You can also erase all the data, and create a new installation. Choose an option and click on “Continue”.



Partitioning the installation

A screen will confirm that you want to make the changes you selected to the system. Click on “Continue”. The next screen will output the partition table on the screen. If you are dual booting, this is where you will know that something is wrong with the installation.

Once you have reviewed the partition table, select “Finish partitioning and write changes to disk”, and click on “Continue”. You will be prompted to choose a root user name and password, as well as a regular user name and password. Fill in these fields, and click on “Continue”. The Operating System should start installing now. Once the installation is complete, a prompt will inform you that the installation is complete. Click on “Continue”, the system should re-boot at this point of time, and the GNOME desktop environment should show up. This is just a skeletal version of the operating system, and does almost nothing out of the box. You will have to install packages using apt-get or the synaptic package manager.



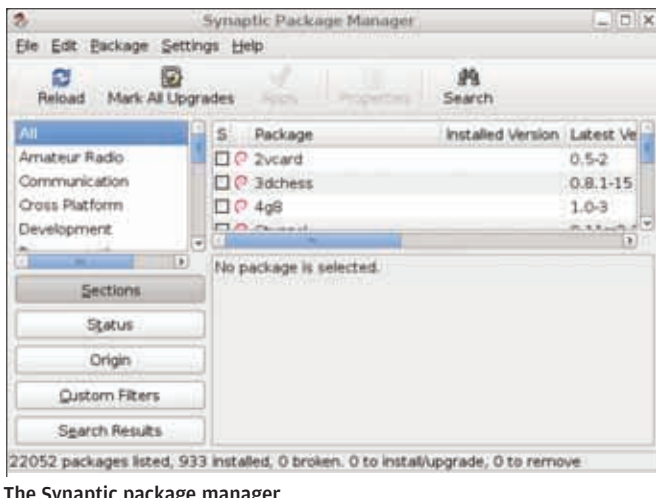
The partition table

5.5.2 Installing Applications

Installing applications is a breeze on Debian, and dependencies are resolved easily. All the packages are pre-compiled, and .deb packages are the most common packages for Linux software on the internet, after tarballs. If you are familiar with Ubuntu, you will be as comfortable with Debian. Just fire up the Synaptic package manager, select a list of software you want to install, and click on “Apply”.

The number one reason for using Debian is the wide support for packages. Debian is a very good option for offices that have to install a Linux distro on a number of machines. Using the 5 DVD version of Debian is a great option here. However, for home users, Linux Mint is a much better choice, with all the benefits of Debian and some more. Linux Mint supports the entire repository of Debian, as well as Ubuntu, and its own repositories.

Debian is also suited for those who plan to use a machine for a long time, say over two years. This is because of the development cycle of Debian, which essentially works as Ubuntu’s LTS (long time support) version. Both Linux Mint and Ubuntu are updated many times in a year, and downloading and installing the newer versions can be a pain, especially if you do not have a great internet connection.



The Synaptic package manager

5.6 Mandriva

Mandriva is a popular distro developed by a company of the same name that has a number of versions for corporate and server use. Mandriva is one of the better looking distros around, and uses the GNOME desktop environment very effectively. Mandriva was earlier known as Mandrake Linux, and originally based on Red Hat. Mandrake Linux was supposed to make Linux easy to use and configure. The second aspect of Mandrake was focused on, which had a detailed control panel which was familiar to what Windows users were used to. The popularity of Mandrake shot up in the early 2000s, and brought with it a legal battle for the company. This was over the name “Mandrake”, which was copyrighted by the Hearst Corporation, who owned the comic book series “Mandrake the Magician”. The name was changed to Mandriva after that. The free edition of Mandriva is to Red Hat what Ubuntu is to Debian, and is still one of the more popular distros around. There are bi-annual releases of Mandriva since 2007, with each release having the year appended to the name instead of the version number. Apart from the yearly releases, there are half-yearly releases, known as the spring editions. The spring editions have a .1 appended to the year name. The latest release of the Operating System is

Mandriva 2009.1. Mandriva is pretty general purpose, and can be used as effectively for multimedia or gaming purposes as well.

5.6.1 Installing Mandriva

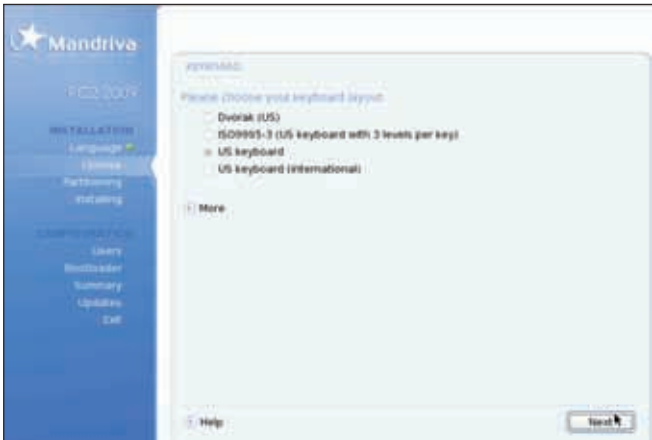
Mandriva has a nice Graphic User Interface, which, like OpenSUSE, uses a wizard-like approach with two sections, the Installation section, and the configuration section. Place the disk in the tray, and restart the system. This is the screen you will see, if you are using the DVD version and not the Live CD:



The startup screen while installing

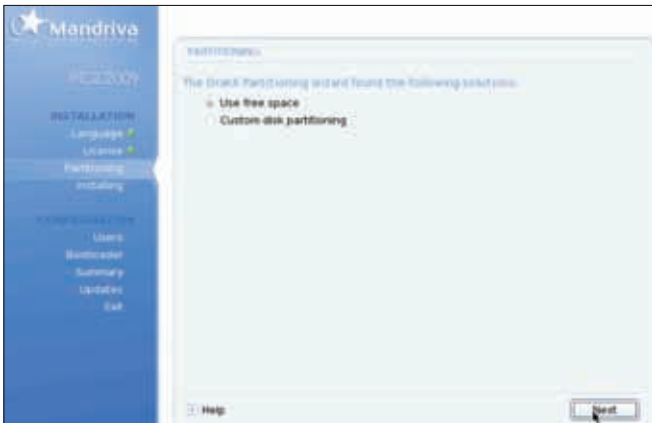
Choose Install Mandriva Linux 2009. You can change the language of the installer by hitting the [F2] key. Next, you will have to choose a language for the operating system, which is not the keyboard map. Asian languages are available, and more importantly, you can choose to install more than one language at once. Click on “Next” when you are done. The next screen is the license agreement and the release notes. Go through the agreement, select Accept, and click on “Next”. This is the keyboard map screen, and there are only 4 options in the default screen (click on more if you use a different map), but these are rather confusing.

The first option is DVORAK, the rest are variations of the



The keyboard layout screen

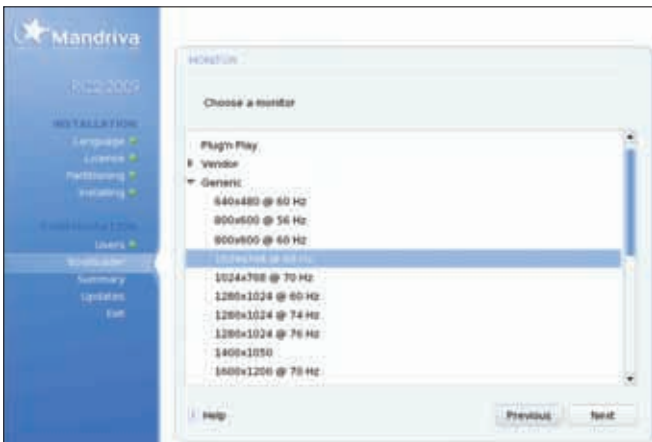
US keyboard map. The right option for most computers is “US Keyboard”, which is the third option. Select this option and click on “Next”. The next screen throws up the partitioning options. By default, Mandriva uses the free space in the hard disk instead of erasing all the data, which is pretty convenient. If you know what you are doing, select the Custom disk partitioning option and click on “Next”, otherwise click on “Use free space”, and click on “Next”.



The DrakX partitioning wizard

The next screen prompts you to copy the contents of the installation medium to the hard drive. Check on the “Copy whole CDs” option and click on “Next”. The next screen is the Package Group Selection, which is a leftover from previous installations of Mandriva. For 2009.1, select “none” and click on next. If you have burnt more than one CD, or is an old installation, select “CD-ROM”. The other options are not relevant for regular users.

The next screen gives you a selection of desktop environments. KDE and GNOME are available, along with a custom option. Choose one of the two, and click on “Next”. The installer will now move from the Installation part to the Configuration part. You will have to set a root password, and a user password. This might seem a little unnecessary for Windows users, but it is one of the many things that makes Linux a more secure operating system in some ways. Click on “Next”, and select a screen resolution for the Bootloader. This depends on the screen resolution of your monitor. Select the appropriate option, and click on “Next”



Choosing a monitor resolution for the Bootloader.

A summary of the installation will show up now. Just click on “Next”. The next screen should prompt you for version updates. The safest option here is “No”, but if you have a LAN

connection with automatic configuration, or is connected to a router, you can select “Yes” for Mandriva to download updates while installation.



Allowing updates while installation

The installation will proceed, and once it is done, click on “Reboot”. You should now see the bootloader with Mandriva, and Mandriva (Safe Mode). If you dual booted the system, the other Operating System should also show up here.

5.6.2 Installing Software

The control center of Mandriva is a great place to start to get to know your system better. Mandriva’s control centre makes Mandriva one of the easiest system to tweak and tune to your preferences, using a well-organised UI. Head over to the software management tab, and select the “Install and remove software” icon to add/remove software.

The command line equivalent is

```
# urpmi (package name)
```

Urpmi is a special variant of rpm, which resolves dependencies during the installation.

Mandriva provides a good range of applications out of the box, and all the essentials are covered. OpenOffice.org, Firefox, Pidgin and Gimp are included, as well as media players and even a DVD playing software. Mandriva on the surface, is a great distro for

those new to Linux to be comfortable with, but underneath, the functionality for advanced users is not compromised on.

5.7 PC-BSD

BSD is a completely different line of operating system, but one that is based on Unix. BSD stands for Berkeley Software Distribution. PC-BSD is a branch from this movement. BSD was put together at the Berkeley university.

For all practical purposes, it works in the same way as Linux. One of the interesting things with PC-BSD is the inclusion of official drivers for hardware from vendors such as Intel and NVIDIA. So if you have an NVIDIA graphics card, this might be something to keep in mind.



The PC-BSD desktop

PC-BSD is easy to use. It's meant for the casual user and it can be used as an everyday operating system.

The installation is simple but the list of packages provided aren't categorised so you need to know what applications you need. The large installation size means you get more by default as compared to the more compact distributions. The KDE environment and community has applications made for all kinds of purposes. For example, there are plenty of educational programs in PC-BSD. You'll notice that there is a strong emphasis on language, science and maths. There are also simple desktop games in the package. While other distributions focus on giving you the bare necessities, PC-BSD has something for everyone.



PCBSD's Welcome Screen

5.7.1 How to install

The installation of BSD distributions is along similar lines as the other Linux distributions. The Welcome screen is very basic and has a list of options to choose from. Type 1 to start the installation procedure. The other options are there for troubleshooting in case you have problems installing PC-BSD



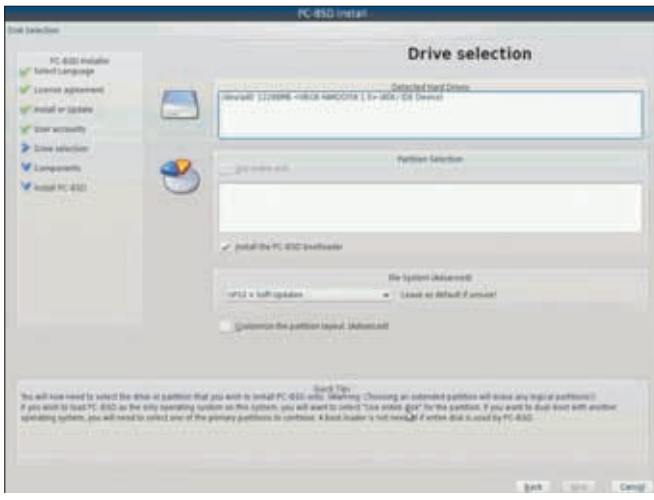
PC-BSD booting for the first time

on your system. They can be used to disable ACPI or other safe modes.

When the installer starts, the first step is to set the language for the installation and the time region. In our case, we select Asia/Calcutta. The next step is a license agreement. The step after that is to choose the kind of install you want to do. If you already have PC-BSD installed, choose the Update/Repair mode. For almost everyone who's installing PC-BSD for the first time,

choose Fresh Install. We'll use the desktop edition and choose to install from CD/DVD/USB. Click Next.

The next step is entering the login information for the account. Enter a password for the administrator account (root). Then, enter the user account name and password. Click Add to finalise it. If you have many other users using the computer, you



Selecting hard drive on which PCBSD is to be installed

can create more Ids similarly. If you want the system to boot up into your account, check the Auto-login user option.

The step after that is the partitioning bit. Here, you can choose if you want the entire hard drive to be used. This is recommended if you have an empty drive have no plans of using any other operating system on your PC. It's also recommended if you have two hard drives and one of them is to be dedicated for PC-BSD. You can customize the partition by clicking the Customize the partition layout checkbox. Create a base partition with mount point `/` which will hold all of your data. The swap partition can be small between 500 MB to 2000 MB in size.

Now, choose the applications you want present in your installation. To do this, select the applications you want from the top list and click the down arrow icon to add them. Click Next to proceed. Unlike other distributions where there

are categories you can choose, PC-BSD specifically asks you the applications you want. This might be good but is a little cumbersome and confusing for new users. If you have the disk space, select all of them and proceed. The installation should then begin. It will take a while as PC-BSD is slightly larger than other CD-based distributions.

Once the installation is complete, the installer will reboot the computer. In the meantime, remove the disc and the bootloader will show you a similar Welcome screen to that of the installation disk. Press 1 to boot or you can wait for 10 seconds before it automatically does. During the first boot, PC-BSD will run a couple of tests and then check your monitor's capabilities. Your screen should flicker a couple of times and you'll be asked to choose the resolution to use.

5.7.1 Interface and Customising

PC-BSD uses KDE as its window manager and environment unlike other Unix based operating systems. There are a few differences between Gnome and KDE. The first and most obvious one is a single Start button and a single bar. The Start menu has section within it for applications and locations. There's also a search function to quickly look for applications from the list.

The second from the left lists the optical discs and external drives connected to your computer. The one after that shows a list of four virtual desktops. You can switch between them. The rest of the space is to place running applications in.

The right end of the bar contains the system tray which has the update manager, clipboard viewer and other system related programs running. You can change the look and functionality of the panel by



The PC-BSD widget menu

clicking the rightmost icon on it. The size of the panel can be changed and widgets can be added to it.

The interesting thing about PC-BSD is the icon on the right top of the desktop which allows users to add widgets to the desktop. This is similar to the widgets found in Windows Vista and Windows 7. Widgets are tiny applications that can be fixed to the desktop. Adding them is simple. Click the icon on the right top and click on Add Widgets. Double click on a widget to install it. The widgets can be moved around the desktop. The options for the widgets can be accessed clicking on the widget itself.

All the system preferences can be accessed by clicking on the Start button and then on System Settings. The default web browser is Konqueror which is something that comes with KDE based distributions. The latest version of Firefox also comes installed and so does Thunderbird, the mail client.

The Computer menu is the equivalent of the Places menu in Gnome distributions. When you right-click on the desktop, the alternate menu is also lets you run applications using the Run command option. Customising the desktop can also be done from here.

5.7.2 Installing applications

Installing applications on PC-BSD is done using the Add/Remove Software utility. It's available under the Applications > System section in the start menu. Preinstalled applications and components can be removed from this program as well.

One of the good things about PC-BSD is the package manager. Where as other open source operating systems require compiling of source packages, PC-BSD works in a similar manner as Windows. Precompiled packages are available and are really simple to install. The format used is .PBI. Doubling clicking on a file starts a wizard-style installation procedure. Uninstalling programs can be done in a similar manner.

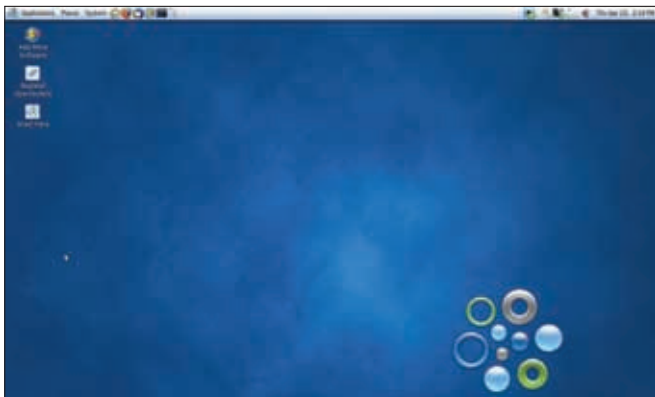
5.7.3 Special applications

There are some fundamental applications and services put into PC-BSD which make it different from other distributions. There are packet filters and firewalls that are a part of OpenBSD. There is a process that fine adjusts CPU depending on the

requirements of applications. This is useful if you run a laptop and can make the battery last longer. The Updater makes updating of all the packages in the distribution really simple as well.

5.8 OpenSolaris

OpenSolaris is yet another project like BSD which is related to the Unix platform. It is based on Sun's Solaris operating system. Future Solaris releases will have technology from the



The OpenSolaris desktop

OpenSolaris distribution. Like Gentoo, OpenSolaris has support for many other platforms such as SPARC and Xeon EM64T.

Just like the compact Linux distributions such as Damn Small Linux, SLAX and Puppy Linux, there are also distributions extracted from OpenSolaris such as MilaX (www.milax.org). Some of the other full-scale distributions are SchilliX (<http://schillix.berlios.de>), Nexenta (www.nexenta.org) and Belenix (www.belenix.org).

OpenSolaris uses the ZFS file system which was developed by Sun for the Solaris. It's a 128-bit file system which can store 18 billion billion times more data than 64-bit file systems.

5.8.1 Installation

Insert the OpenSolaris disc and boot into the interface. Double click the Install OpenSolaris icon to start the installation process. The screen after the Welcome screen is the partitioning screen.

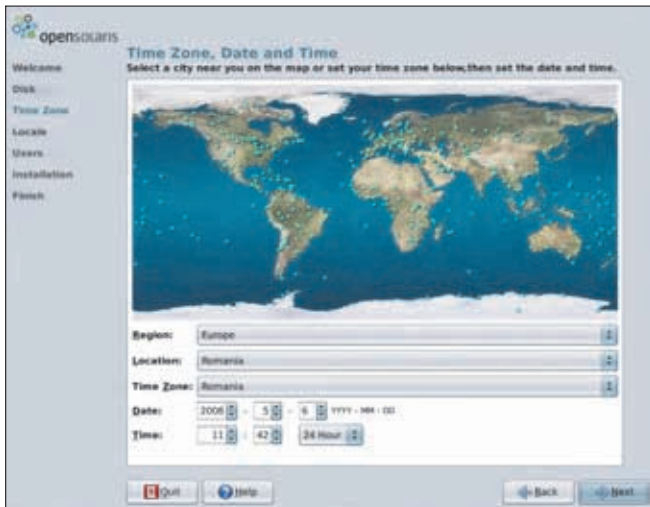
You can select the Use the whole disk option to use the entire hard drive for the operating system. To create new partitions, click on the first dropdown menu and enter a size for the partition.



Partitioning the disk for OpenSolaris

In the next screen, enter your location, timezone, date and time. After that, select the language you want to use.

The screen after that is to enter the root password entry and the other user account. The next step displays an overview of all the settings. Click on Next to star the installation. At the end of the installation, click on Reboot.



Selecting the timezone and current date

When OpenSolaris boots, enter the username and password and you're ready to go.

5.8.2 Applications

One of the specialities of the new OpenSolaris operating system is the easy deployment of the operating to different computers on a desktop using an application called Crossbow. The file system used by OpenSolaris is called Image Packaging System. The web installation feature makes installation of applications a breeze. A single click on a web page lets you install application without having to download and then compile like other distributions.

Some of the other improvements in the latest build are support for more hardware including the Nehalem based Xeon processors. There is also improved support for virtualization servers.

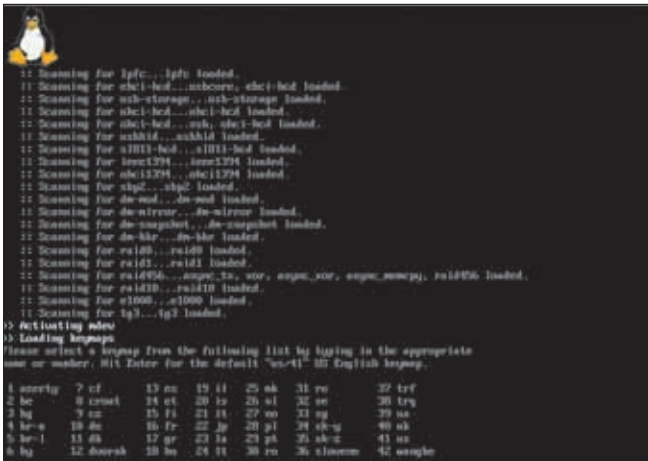
5.9 Gentoo

Development on Gentoo began close to a decade ago. Gentoo Linux is a distribution which supports some of the widest range of system architectures. Other than x86, there are also builds for PowerPC, IA64, SPARC, PowerPC and even the Sony Playstation 3's Cell processor. Saboyan Linux and SystemRescueCD are two well known distributions that are extracted from Gentoo.

Gentoo Linux is a distribution that many fear and some others love. Many would say it's supposedly meant for the more enthusiastic lot. It is known to be highly customisable and the user gets complete control over the operating system. This would also mean that those installing Gentoo for the first time might find it a difficult to use. For someone who has graduated from Ubuntu to another distribution and then on to Gentoo, it's considered to be at the pinnacle.

5.9.1 Installation

The installer isn't as difficult as is expected. With the CD popped in, you simply press Enter to start the installation. Just before the user interface loads, you'll be asked to enter the keyboard layout. Press Enter to continue at this point. The interface will then load and you'll notice that it looks a lot different than the other KDE and Gnome based environments. There are two ways to install Gentoo – one using the user



Selecting the key layout for Gentoo

interface or the terminal based. The GTK interface one works just fine unless you have a really low-end configuration.

The first step in the installation is the partitioning. You can allow Gentoo to setup partitions for you but it's best if

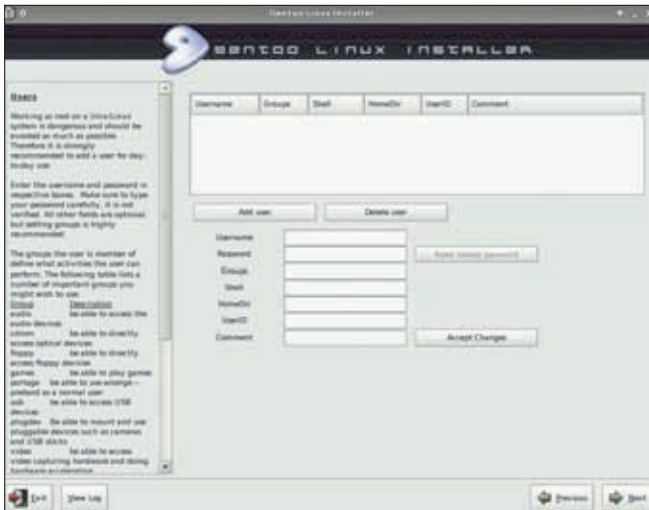


Creating partitions under Gentoo

you manually do so if you have other partitions and operating systems on the drive. Creating partitions is simple. A large ext3 partition and a 500 MB to 2000 MB swap partition is recommended. A small /boot partition is also recommended. Other distributions ask you to select the mount point for the partitions during the creation itself. In Gentoo, the next step is meant for that very reason. Set your largest partition's point mount to / and the small partition to /boot. The swap partition requires no mount point.

The installation should then start. At the end of the files being installed, you'll be asked to enter a root password and then the timezone. The next step is configuring the network adaptor. Select whether your network uses DHCP settings or a manual IP is to be allotted. Click on Save and click on Next. Some additional packages will then be installed.

Your login information for Gentoo needs to be entered after this. You can add multiple accounts in the same manner and also include the path to your home folder. Next, the list of packages you want to installed must be selected. Gentoo has the applications put into neat categories into a single page. Select the ones you need and click Next. The installation process will then begin.



Creating user accounts in Gentoo

Once Gentoo is completely installed, you're asked to select the service you want to load during startup. Think of it like Windows services you don't need that you shut down to save memory. The last step is selecting the window manager and other system components. There are quite a few alternatives that come installed with Gentoo. You need to choose which ones. Fluxbox is a great window manager to try out if you're bored with the usual ones. It's light, minimalistic, highly configurable and it looks very attractive. After this, exit the installer and reboot the computer.

When Gentoo boots for the first time, you'll have to login using the root username and password that you set during the installation. Use the command `startx` to start Xwindows.

5.10 Fedora

Fedora is a distribution funded by Red Hat. Fedora is based completely on Redhat and so uses the traditional RPM based package management system. It is known to be ranked right next to Ubuntu in terms of popularity. Fedora uses the KDE desktop environment system. Fedora was created when Red Hat became Red hat Enterprise Edition and the home users distribution was abandoned.

The latest builds of Fedora come with KDE and Gnome. The distribution is available on CDs and also on DVDs. It comes with everything that a casual desktop user needs. Just like Ubuntu, Fedora too doesn't come with the codecs required to play MP3 audio formats. Many free software also available for Fedora aren't bundled with the distribution. In general, Fedora sticks to free and opensource software.

5.10.1 Installation

To install Fedora, set the DVD drive to bootable in the BIOS and insert the disc into it. When the interface loads up, select the language for the install and then



Setting up the network configuration

the keyboard layout.

Enter the network name for the PC. This will be the named by other computers to access this Fedora PC. Next, select your timezone from the map or use the drop down menu and choose your location from the list.

Next, enter the password for the root account. The partitioning screen is next in line. You can choose to either use the entire drive or remove the current

Linux distribution or even make do with the empty space available. If you have multiple drives connected to your PC, you can choose the drive you want to install on using the dropdown menu at the bottom of the screen. Click Next. Click the Write changes to disk button to make the changes to the partition changes permanent.

Next, select the category of software is to be selected. For most purposes, just selecting the Office and Productivity section. Click Next and the installation will install all the necessary files to the drive. The next few steps will be for the creation of the user accounts. The installation completes.



Creating a custom partition layout

5.11 Granular Linux

Granular Linux is an all-purpose operating system built in India and is one of the newer distributions to hit the internet. It is based on the PCLinuxOS distribution. Not too long back, a DVD version of the distribution was released which included games along with some other applications. Granular also comes with official ATI and NVIDIA drivers.

One of the packages that was provided on the DVD was the LG3D which is called Project Looking Glass. This application gives users a much more interactive and visual appealing looking interface.

5.11.1 Installation

Insert the Granular installation disc and boot the system. Like some other distributions, there are tests to ensure that there are no faults with the hardware or the installation disc. To install, use the Granular option and press Enter. The installer will load and you'll need to set the keyboard layout to US keyboard.

The next screen asks you to enter the login username and password. Both the values are root. The Granular desktop will then boot. Double-click the Install Granular icon to start the installation process.



Initial login into the installation screen

DrakX is the partitioning tool used by Granular. Click the Use free space if you have a dedicated hard drive for the installation otherwise use the Custom disk partitioning option. You'll also see a warning window asking you to backup all of your data before the installation.

You need to create two partitions – an ext3 partition for the operating system and the data. The other partition has to be a Linux swap partition. Click the Create button to do this. If you have the entire hard drive to use with Granular Linux, simply click Auto allocate and the partitions will be created automatically.

Granular doesn't give any further options to choose the packages and categories of software. Proceeding will start the installation.

At the end of the installation, you'll be prompted to enter information for the boot loader. You can choose the interface of the boot loader and the hard drive on which you want it to be installed. There is also a delay setting which tells the boot loader how long to wait before booting the operating system.

When Granular reboots, you'll have to enter authentication information for the machine. This include the administrator passwords and the user account. You can also select a login image for your username. You'll see that the login interface is somewhat similar to Windows.

5.11.2 Interface and Customising

Granular Linux runs on KDE so a lot of the features and layout of items will seem similar. The menu bar sits on the top of the screen. A list of program shortcuts are on the top right before the virtual desktops list. You'll also find Enlightenment as a part of Granular.

5.11.3 Applications

The Granular Control Center is an important part of the OS. It's a system configuration tool which lets you set up everything from hardware to file sharing on a network. It's visible as a tools icon on the menubar. All the tasks possible are available in the left panel.

Because PCLinuxOS was derived out of Mandriva, Granular too uses the RPM file installation format. It too uses the Synaptic Package Manager. Granular comes with the mklivecd script which allows users to create their own versions of the distribution or even a LiveCD.

Other important Linux Distros

The detailed install guidelines above should make you comfortable with installing most Linux distros. There are a few derivative distros, and distros used for particular purposes. The list of distros are preferred by their users depending on how they are used, or how they function. After trying out their first distro, most Linux enthusiasts cycle through a wide range of distros in an attempt to find the one most suitable to them. In fact, many do this habitually, going on to the latest distro available, just because of some improvements in the releases of the other distros. This short guide should help you select the pick of the lot for your needs:

5.12 Other important Linux distributions

5.12.1 Arch Linux

Arch Linux is one of those distros that follow the KISS principle. Arch is similar to Slackware in many ways, but has a few improvements. If you are installing Arch Linux, refer to the Slackware installation guide, as the two are pretty similar. Arch Linux has rolling updates, which are more frequent than Fedora or Gentoo, and hence require a constant Internet connection. Arch Linux is geared towards engineers, coders or command line enthusiasts. Arch Linux has a large repository, and the

dependencies are sufficiently resolved using Pacman, the installer. A lot of tools and applications in Arch Linux do not have a default GUI, but the code itself is well documented, and with easy instructions for new users to understand. The latest Version is Arch 2009.02, and Arch seems to be following the Mandriva naming convention, starting from this release.

5.12.2 Linux Mint

Linux Mint has a great development team, a dedicated and helpful IRC channel, and a whole bunch of tricks up its sleeve to ease the transition from Windows to Linux for new users. In fact, we recommend that Linux Mint be the first distro that you try out, for a number of reasons. First of all, Linux Mint is based on Ubuntu, so it is compatible with the repositories of Linux Mint, Ubuntu as well as Debian, which is a wide range of software that is compatible with the distro. A whole bunch of video and audio codecs are available out of the box, and even obscure document formats such as jpg images packed in a rar file or comic book formats are opened by the document reader by default. Linux Mint, in fact, offers more document support out of the box than the latest release of Windows 7 does. One surprising little functionality is the ease with which Linux Mint can interface over WiFi. WiFi functionality is present out of the box, which is still very rare in Linux distros.

Linux Mint also takes efforts to look great, and along with Sabayon and Mandriva, is one of the neatest distros that we have seen. Every time you start the command line, instead of a boring prompt, there is some ASCII art and a joke. Linux Mint basically takes the weight of learning a new Operating System, and one meant for the 1337 crowd off your shoulders, and makes you feel comfortable from the word go. There are also a small bunch of tools, mostly written in python that comes with Linux Mint. These tools are small extras for the user, and are pretty useful. One of the tools, for example, can transfer files from one Mint Install to another over the same network. Refer to the Ubuntu installer for installing Linux mint.

5.12.3 SLAX

As compared to the likes of Ubuntu, SLAX might not be as well known. It's a special distribution mainly because of its size. The

disc image is just 190 MB. The disc is a LiveCD which means you don't have to install it as the operating system runs off it.

There are users who need to travel a lot and carry their work with them. Carrying just the data isn't sufficient. People have specific applications for their work and those have to go along as well but it isn't possible always. For this reasons, people have started carrying their operating systems on their portable drives. Linux and the variants are perfect for this kind of requirement. SLAX is an ideal distribution to run on a portable drive. You can set it up to run along with Virtualbox Portable. The latest version runs KDE 3.5 and has the very traditional KDE layout and look to it. There is a package manager which lets you plug in additional features and components. If you've wanted something a little more thorough than Damn Small Linux, then SLAX is for you.

5.12.4 Multimedia Distros

There are a whole range of multimedia distros. These are the XBMC media centre distro, Dyne:bolic, which is geared towards editing and GeexBox. XBMC worked great, is a nice Live CD to show off on, but is not too useful beyond that. Is a great interface for a dedicated HTPC, if you are willing to use a machine in that way. Dyne:bolic looks outdated, and comes without the codecs, which is a bit thick for a multimedia editing distro. GeexBox looks like a kid's application, and is a live CD without an easy installer. It does not come with a fair few codecs pre-installed, so unless you are willing to take the time and effort to compile GeexBox with the codecs you want, or install it on an HTPC and improve it (XBMC is better for that), it is not a great idea to use this one. For multimedia capabilities on a regular desktop, we reiterate our recommendation of Linux Mint.